Forward LTL_f Synthesis: DPLL At Work

Marco Favorito
Banca d'Italia
marco.favorito@gmail.com

Abstract

This paper proposes a new AND-OR graph search framework for synthesis of Linear Temporal Logic on finite traces (LTL_f), that overcomes some limitations of previous approaches. Within such framework, I devise a procedure inspired by the Davis-Putnam-Logemann-Loveland (DPLL) algorithm to generate the next available agent-environment moves in a truly depth-first fashion, possibly avoiding exhaustive enumeration or costly compilations. I also propose a novel equivalence check for search nodes based on syntactic equivalence of state formulas. Since the resulting procedure is not guaranteed to terminate, I identify a stopping condition to abort execution and restart the search with state-equivalence checking based on Binary Decision Diagrams (BDD), which I show to be correct. The experimental results show that in many cases the proposed techniques outperform other state-of-the-art approaches.

1 Introduction

Program synthesis is the task of finding a program that provably satisfies a given high-level formal specification [Church, 1963]. A commonly used logic for program synthesis is Linear Temporal Logic (LTL) [Pnueli, 1977; Pnueli and Rosner, 1989], typically used also in model checking [Baier and Katoen, 2008]. LTL *on finite traces* (LTL_f) [De Giacomo and Vardi, 2013], a variant of LTL to specify *finite*-horizon temporal properties, has been recently proposed as specification language for temporal synthesis [De Giacomo and Vardi, 2015]. The LTL_f synthesis setting considers a set of variables controllable by the agent, a (disjoint) set of variables controlled by the environment, and a LTL_f specification that specifies which finite traces over such variables are desirable. The problem of LTL_f synthesis consists of finding a finite-state controller (i.e. the program) that at every time step, given the values of the environment variables in the history so far, sets the next values for each agent proposition such that the generated traces comply with the LTL_f specification.

The basic technique for solving LTL_f synthesis amounts to constructing a deterministic finite automaton (DFA) corresponding to the LTL_f specification, and then considering it as a game arena where the agent tries to get to an accepting state regardless of the environment's moves. Then, a *winning strategy*, i.e. a finite controller returned by the synthesis procedure, can be obtained through a backward fixpoint computation for *adversarial reachability* of the DFA accepting state.

Related works. State-of-the-art tools such as Lydia [De Giacomo and Favorito, 2021] and Lisa [Bansal *et al.*, 2020a] are based on the classical approach. The main drawback of this technique is that it requires to compute the entire DFA of the LTL_f specification, which in the worst case can be doubly exponential in the size of the formula. Therefore, the DFA construction step becomes the main bottleneck.

A natural idea is to consider a forward search approach that expands the arena on-the-fly while searching for a solution, possibly avoiding the construction of the entire arena. Forward-based approaches are at the core of the best solution methods designed for other AI problems: Planning with fully observable non-deterministic domains (FOND) [Ghallab *et al.*, 2004; Geffner and Bonet, 2013; Cimatti *et al.*, 1998; Cimatti *et al.*, 2003], where the agent has to reach the goal, despite that the environment may choose adversarially the effects of the agent actions, and Planning in partially observable nondeterministic domains (POND), also known as *contingent planning*, where the search procedure must be performed over the *belief-states* [Reif, 1984; Goldman and Boddy, 1996; Bertoli *et al.*, 2006]. However, techniques developed for such problems cannot be applied to ours: in a FOND planning problem, represented with PDDL [Haslum *et al.*, 2019], the search space is at most single-exponential [Rintanen, 2004], whereas for LTL_f synthesis the state space can be of double-exponential size wrt the size fo the formula; in a POND planning problem, despite the double-exponential size of the state space, belief-states have a specific structure [Bertoli *et al.*, 2006; Thanh To *et al.*, 2009], and therefore techniques for solving it cannot be directly applied to LTL_f synthesis.

For these reasons, researchers have been looking into forward search techniques specifically conceived for solving LTL_f synthesis. Two notable attempts in this direction have been presented in [Xiao *et al.*, 2021] and [De Giacomo *et al.*, 2022]. The former work presents an on-the-fly synthesis approach via conducting a so-called Transition-based Deterministic Finite Automata (TDFA) game, where the acceptance condition is defined on transitions, instead of states. The main issue of that approach is the full enumeration of agent-environment moves, which are exponentially many in the number of variables. Moreover, due to the fact that the acceptance condition is defined on transitions, every generated transition has to be checked for acceptance. The latter work instead proposes a search framework for LTL_f synthesis, where the DFA arena is seen as an AND-OR graph, and the available moves are found according to the formula associated to the current search node, by means of a Knowledge Compilation (KC) technique: Sentential Decision Diagrams (SDD) [Darwiche, 2011]. Notably, they are able to branch on propositional formulas, representing several evaluations, instead of individual ones. This can drastically reduce the branching factor. Nevertheless, for certain types of problem instances, the approach can get stuck with demanding compilations of the state formulas, needed *both* for state equivalence checking and for search node expansion. Moreover, the requirement of having irreducible representation of agent-env moves can be of little usefulness if the branching factor of the search problem is already high, hence resulting in an even greater compilation overhead.

Contributions. I think there is the need of a search approach that scales well with the increase of computational power, and that uses such power for actually exploring the search space, rather than wasting time either slavishly enumerating the exponentially many variable assignments, or by finding the minimal representation of the available search moves. My contributions are the following. First, I identify limitations of the previous AND-OR graph search framework, based on the EXPAND function, and propose a more general and versatile search framework for LTL_f synthesis, based on two primitive operations: state-equivalence checking and search node expansion. Then, I propose two realizations of these operations in the context of LTL_f synthesis: one is a search graph expansion technique based on a procedure inspired by the famous Davis-Putnam-Logemann-Loveland (DPLL) algorithm; and the other is a state-equivalence checking technique based on structural equivalence of state formulas. Unfortunately, the resulting search algorithm does not terminate in general, but I designed a stopping condition to abort execution and resort to the KC-based state-equivalence checking using Binary Decision Diagrams (BDD) [Bryant, 1992], that I show to be correct. Finally, I describe my implementation in a new tool called Nike, and compare its performance on known benchmarks with other state-of-the-art tools, showing its surprising effectiveness.

2 Preliminaries

LTL_f Basics. Linear Temporal Logic over finite traces, called LTL_f [De Giacomo and Vardi, 2013] is a variant of Linear Temporal Logic (LTL) [Baier and Katoen, 2008] that is interpreted over finite traces rather than infinite traces (as in LTL). Given a set of propositions $\mathcal{P}$, the syntax of LTL_f is identical to LTL, and defined as (wlog, we require LTL_f formulas are in Negation Normal Form (NNF), i.e., negations only occur in front of atomic propositions): $\varphi ::= tt \mid ff \mid p \mid \neg p \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \circ\varphi \mid \bullet\varphi \mid \varphi_1 \mathcal{U} \varphi_2 \mid \varphi_1 \mathcal{R} \varphi_2$. tt is always true, ff is always false; $p \in \mathcal{P}$ is an *atom*, and $\neg p$ is a *negated atom* (a literal l is an atom or the negation of an atom); $\wedge$ (And) and $\vee$ (Or) are the Boolean connectives; and $\circ$ (Next), $\bullet$ (Weak Next), $\mathcal{U}$ (Until) and $\mathcal{R}$ (Release) are temporal connectives. We use the usual abbreviations $true \equiv p \vee \neg p$, $false \equiv p \wedge \neg p$, $\Diamond\varphi \equiv true\, \mathcal{U}\, \varphi$ and $\Box\varphi \equiv false\, \mathcal{R}\, \varphi$. Also for convenience we consider traces $\rho \in (2^{\mathcal{P}})^*$, i.e., we consider also empty traces ϵ as in [Brafman *et al.*, 2018]. More specifically, a trace $\rho = \rho[0], \rho[1], \ldots \in (2^{\mathcal{P}})^*$ is a finite sequence, where $\rho[i]$ $(0 \leq i < |\rho|)$ denotes the i-th interpretation of ρ, which can be considered as the set of propositions that are $true$ at instant i, and $|\rho|$ represents the length of ρ. We have that $\epsilon \models \varphi$ if φ is tt, an $\mathcal{R}$-formula or $\bullet$-formula, hence $\epsilon \models \Box false$. $\epsilon \not\models \varphi$ if φ is ff, a literal, $\mathcal{U}$-formula or $\circ$-formula, hence $\epsilon \not\models \Diamond true$. We consider the semantics of LTL_f as presented in [Brafman *et al.*, 2018].

We denote by $\mathsf{cl}(\varphi)$ the set of subformulas of φ, including tt and ff. We denote by $\mathsf{pa}(\varphi) \subseteq \mathsf{cl}(\varphi)$ the set of literals and temporal subformulas of φ whose primary connective is temporal [Li *et al.*, 2019]. Formally, for an LTL_f formula φ in NNF, we have $\mathsf{pa}(\varphi) = \{\varphi\}$ if φ is a literal or temporal formula; and $\mathsf{pa}(\varphi) = \mathsf{pa}(\varphi_1) \cup \mathsf{pa}(\varphi_2)$ if $\varphi = (\varphi_1 \wedge \varphi_2)$ or $\varphi = (\varphi_1 \vee \varphi_2)$. Having LTL_f formula φ, replacing every temporal formula $\psi \in \mathsf{pa}(\varphi)$ with a propositional variable a_ψ gives us a propositional formula φ^p; we call this operation *propositionalization of* φ. Note that $\varphi^p \in \mathcal{B}^+(\mathsf{cl}(\varphi))$, i.e. φ^p is a positive Boolean formula over variables $\mathsf{cl}(\varphi)$. Let $\phi = \varphi^p$, we denote with $\phi^{\mathsf{tf}} = \varphi$ the inverse operation of $\cdot^p$. Two formulas φ_1 and φ_2 are propositionally equivalent, denoted by $\varphi_1 \sim_p \varphi_2$, if, $C \models \varphi_1^p \leftrightarrow C \models \varphi_2^p$ holds for every propositional assignment $C \in 2^{\mathsf{pa}(\varphi_1) \cup \mathsf{pa}(\varphi_2)}$.

An LTL_f formula φ is in neXt Normal Form (XNF) if $\mathsf{pa}(\varphi)$ only includes literals, $\circ$- and $\bullet$-formulas. For an LTL_f formula φ in NNF, we can obtain its XNF by transformation function $\mathsf{xnf}(\varphi)$, defined as follows:
- $\mathsf{xnf}(\varphi) = \varphi$ if φ is a literal, $\Box false$, $\Diamond true$, $\circ$-, $\bullet$-formula;
- $\mathsf{xnf}(\varphi_1 \wedge \varphi_2) = \mathsf{xnf}(\varphi_1) \wedge \mathsf{xnf}(\varphi_2)$;
- $\mathsf{xnf}(\varphi_1 \vee \varphi_2) = \mathsf{xnf}(\varphi_1) \vee \mathsf{xnf}(\varphi_2)$;
- $\mathsf{xnf}(\varphi_1 \mathcal{U} \varphi_2) = (\mathsf{xnf}(\varphi_2) \wedge \Diamond true) \vee (\mathsf{xnf}(\varphi_1) \wedge \circ(\varphi_1 \mathcal{U} \varphi_2))$;
- $\mathsf{xnf}(\varphi_1 \mathcal{R} \varphi_2) = (\mathsf{xnf}(\varphi_2) \vee \Box false) \wedge (\mathsf{xnf}(\varphi_1) \vee \bullet(\varphi_1 \mathcal{R} \varphi_2))$.

Note that $\Diamond true$ (resp. $\Box false$) guarantees that empty trace is not (resp. is) accepted by $\mathcal{U}$-formulas (resp. $\mathcal{R}$-formulas).

Theorem 1 ([Li *et al.*, 2019])**.** *Every LTL_f formula φ in NNF can be converted, with linear time in the formula size, to an equivalent formula in XNF, denoted by $\mathsf{xnf}(\varphi)$.*

LTL_f Formula Progression [De Giacomo *et al.*, 2022]. Consider an LTL_f formula φ over $\mathcal{P}$ and a finite trace $\rho = \rho[0], \rho[1], \ldots \in (2^{\mathcal{P}})^*$, in order to have $\rho \models \varphi$, we can start from φ, progress or push φ through ρ. The idea behind *formula progression* is to split an LTL_f formula φ into a requirement about *now* $\rho[i]$, which can be checked straightaway, and a requirement about the future that has to hold on the yet unavailable suffix. That is to say, formula progression looks at $\rho[i]$ and φ, and progresses a new formula $\mathsf{fp}(\varphi, \rho[i])$ such that $\rho, i \models \varphi$ iff $\rho, i+1 \models \mathsf{fp}(\varphi, \rho[i])$. This procedure is analogous to DFA reading trace ρ, where reaching accepting states is essentially achieved by taking one transition after another. Formula progression has been studied in prior work, cf. [Emerson, 1990; Bacchus and Kabanza, 1998]. Here we use the formalization provided in [De Giacomo *et al.*, 2022].

Note that, since ρ is a finite trace, it is necessary to clarify when the trace ends. To do so, two new formulas are introduced: $\Box false$ and $\Diamond true$, which, intuitively, refer to *finite trace ends* and *finite trace not ends*, respectively. For simplicity, we enrich $\mathsf{cl}(\varphi)$, the set of proper subformulas of φ, to include them such that $\mathsf{cl}(\varphi)$ is reloaded as $\mathsf{cl}(\varphi) \cup \mathsf{cl}(\Diamond true) \cup \mathsf{cl}(\Box false)$.

For an LTL$_f$ formula φ in NNF, the *progression function* $\mathsf{fp}(\varphi, \sigma)$, where $\sigma \in 2^{\mathcal{P}}$, is defined as follows:

- $\mathsf{fp}(tt, \sigma) = tt$ and $\mathsf{fp}(ff, \sigma) = ff$;
- $\mathsf{fp}(p, \sigma) = tt$ if $p \in \sigma$, otherwise ff;
- $\mathsf{fp}(\neg p, \sigma) = tt$ if $p \notin \sigma$, otherwise ff;
- $\mathsf{fp}(\varphi_1 \wedge \varphi_2, \sigma) = \mathsf{fp}(\varphi_1, \sigma) \wedge \mathsf{fp}(\varphi_2, \sigma)$;
- $\mathsf{fp}(\varphi_1 \vee \varphi_2, \sigma) = \mathsf{fp}(\varphi_1, \sigma) \vee \mathsf{fp}(\varphi_2, \sigma)$;
- $\mathsf{fp}(\circ\varphi, \sigma) = \varphi \wedge \Diamond true$;
- $\mathsf{fp}(\bullet\varphi, \sigma) = \varphi \vee \Box false$;
- $\mathsf{fp}(\varphi_1 \mathcal{U} \varphi_2, \sigma) = \mathsf{fp}(\varphi_2, \sigma) \vee (\mathsf{fp}(\varphi_1, \sigma) \wedge \mathsf{fp}(\circ(\varphi_1 \mathcal{U} \varphi_2), \sigma))$;
- $\mathsf{fp}(\varphi_1 \mathcal{R} \varphi_2, \sigma) = \mathsf{fp}(\varphi_2, \sigma) \wedge (\mathsf{fp}(\varphi_1, \sigma) \vee \mathsf{fp}(\bullet(\varphi_1 \mathcal{R} \varphi_2), \sigma))$.

Note that $\mathsf{fp}(\varphi, \sigma)$ is a positive Boolean formula on $\mathsf{cl}(\varphi)$, i.e., $\mathsf{fp}(\varphi, \sigma) \in \mathcal{B}^+(\mathsf{cl}(\varphi))$. The following two propositions show that $\mathsf{fp}(\varphi, \sigma)$ strictly follows LTL$_f$ semantics and retains the propositional behavior of LTL$_f$ formulas.

Proposition 1. *Let φ be an LTL$_f$ formula over $\mathcal{P}$ in NNF, ρ be a finite nonempty trace, $\mathsf{fp}(\varphi, \sigma)$ be as above. We have that $\rho, i \models \varphi$ iff $\rho, i+1 \models \mathsf{fp}(\varphi, \rho[i])$.*

Proposition 2. *Let φ and ψ be two LTL$_f$ formulas over $\mathcal{P}$ in NNF s.t. $\varphi \sim_p \psi$, and $\sigma \in 2^{\mathcal{P}}$. Then $\mathsf{fp}(\varphi, \sigma) \sim_p \mathsf{fp}(\psi, \sigma)$ holds.*

We generalize LTL$_f$ formula progression from single instants to finite traces by defining $\mathsf{fp}(\varphi, \epsilon) = \varphi$, and $\mathsf{fp}(\varphi, \sigma u) = \mathsf{fp}(\varphi, \sigma u) = \mathsf{fp}(\mathsf{fp}(\varphi, \sigma), u)$, where $\sigma \in 2^{\mathcal{P}}$ and $u \in (2^{\mathcal{P}})^*$.

Proposition 3. *Let φ be an LTL$_f$ formula over $\mathcal{P}$ in NNF, ρ be a finite trace. We have that $\rho \models \varphi$ iff $\epsilon \models \mathsf{fp}(\varphi, \rho)$.*

We take the definition of the *remove-next* function RMNEXT from [De Giacomo *et al.*, 2022], defined over propositionalized LTL$_f$ formulas in XNF, φ^p:

- $\textsc{RmNext}(\Diamond true) = tt$, $\textsc{RmNext}(\Box false) = ff$
- $\textsc{RmNext}(\varphi_1 \wedge \varphi_2) = \textsc{RmNext}(\varphi_1) \wedge \textsc{RmNext}(\varphi_2)$
- $\textsc{RmNext}(\varphi_1 \vee \varphi_2) = \textsc{RmNext}(\varphi_1) \vee \textsc{RmNext}(\varphi_2)$
- $\textsc{RmNext}(\circ\varphi) = \varphi \wedge \Diamond true$, $\textsc{RmNext}(\bullet\varphi) = \varphi \vee \Box false$

Note that RMNEXT applies to neither $\mathcal{U}$-,$\mathcal{R}$- formulas, since they do not appear in XNF, nor literals (p, $\neg p$), as the input of the function is a propositionalized LTL$_f$ formula in XNF form. We have the following proposition:

Proposition 4. *Given an LTL$_f$ formula φ in NNF, $\forall \sigma \in 2^{\mathcal{P}}, \mathsf{fp}(\varphi, \sigma) \equiv \textsc{RmNext}(\mathsf{xnf}(\varphi)^p|_\sigma)$, where $\mathsf{xnf}(\varphi)^p|_\sigma$ stands for evaluating σ on $\mathsf{xnf}(\varphi)^p$.*

LTL$_f$ Synthesis The problem of LTL$_f$ synthesis is described as a tuple $(\varphi, \mathcal{X}, \mathcal{Y})$, where φ is an LTL$_f$ formula over $\mathcal{X} \cup \mathcal{Y}$, and $\mathcal{X}, \mathcal{Y}$ are two disjoint sets of variables controlled by the *environment* and the *agent*, respectively.

Definition 1. *The* synthesis *problem $(\varphi, \mathcal{X}, \mathcal{Y})$ aims to computing a strategy $g : (2^{\mathcal{X}})^* \to 2^{\mathcal{Y}}$, such that for an arbitrary infinite sequence $\lambda = X_0, X_1, \ldots \in (2^{\mathcal{X}})^\omega$, we can find $k \geq 0$ such that $\rho^k \models \varphi$, where $\rho^k = (X_0 \cup g(\epsilon)), (X_1 \cup g(X_0)), \ldots, (X_k \cup g(X_0, X_1, \ldots, X_{k-1}))$. If such a strategy does not exist, then φ is unrealizable.*

Algorithm 1 SDD-based Forward Synthesis [De Giacomo *et al.*, 2022]

```
1:  function SYNTHESIS(φ) return strategy
2:      if ISACCEPTING(φ) then
3:          ADDTOSTRATEGY(φ, true)
4:          return GETSTRATEGY()
5:      INITIALGRAPH(φ)
6:      n := GETGRAPHROOT()
7:      found := SEARCH(n, ∅)
8:      if found then return GETSTRATEGY()
9:      return EMPTYSTRATEGY() ▷ φ is unrealizable

10: function SEARCH(n, path) return True/False
11:     if ISSUCCESSNODE(n) then return True
12:     if ISFAILURENODE(n) then return False
13:     if INPATH(n, path) then ▷ We found a loop
14:         TAGLOOP(n) return False
15:     ψ :=FORMULAOFNODE(n)
16:     if ISACCEPTING(ψ) then
17:         TAGSUCCESSNODE(n)
18:         ADDTOSTRATEGY(ψ, true)
19:         return True
20:     EXPAND (n) ▷ Uses SDD to partition ψ wrt 𝒴 and 𝒳
21:     for (act, AndNd) ∈GETORARCS(n) do
22:         for (resp, succ) ∈GETANDARCS(AndNd) do
23:             found :=SEARCH(succ, [path|n])
24:             if ¬found then Break
25:         if found then
26:             TAGSUCCESSNODE(n)
27:             ADDTOSTRATEGY(ψ, act)
28:             if ISTAGLOOP(n) then
29:                 BACKPROP(n)
30:             return True
31:     TAGFAILURENODE(n)
32:     return False
```

LTL$_f$ synthesis can be solved by reducing to an adversarial reachability game on the corresponding Deterministic Finite Automaton (DFA) [De Giacomo and Vardi, 2015]. Hence, a strategy can also be represented as a finite-state controller $g : \mathcal{S} \mapsto 2^{\mathcal{Y}}$, where $\mathcal{S}$ denotes the state space of the DFA.

3 Limitations of Previous Works

The motivations for this work lie on the limitations of previous forward LTL$_f$ synthesis approaches namely Xiao *et al.*'s and De Giacomo *et al.* works. Since the implementation of the former approach (Ltlfsyn) has been considered superseded by the latter (Cynthia) in terms of performance, here I focus on Cynthia, although my arguments can be considered more general and not just applicable to specific techniques.

The state-of-the-art forward technique [De Giacomo *et al.*, 2022], implemented in the tool Cynthia, is described by the pseudocode in Algorithm 1. The algorithm is basically a top-down, depth-first traversal of the AND-OR graph induced by the on-the-fly DFA construction, proceeding forward from the initial state, and excluding strategies that lead to loops. The forward-based generation of the AND-OR graph is based on formula progression and on an abstract EXPAND function (Line 20) that, taken in input a search node n, it produces the next available actions and successor states. The presence of loops must be carefully handled; when a loop is detected at node n, the procedure returns false, temporarily considering n as a failure node. Note that node n is not tagged as failure, since it is unknown whether all the or-arcs of n are explored. If later during the search n is discovered as a success node, such information must be propagated from

n backwards to the ancestor nodes of n. It should be noted that, in a forward search on an AND-OR graph, it is critical to handle loops with the assistance of this backward propagation, implemented in BACKPROP (Line 29), as illustrated in [Scutellà, 1990]. For more details on the search algorithm, please refer to the original paper [De Giacomo *et al.*, 2022].

The realization of the abstract EXPAND function was based on Sentential Decision Diagrams (SDDs) [Darwiche, 2011]. The SDDs have been used for two subtasks: (*i*) **state-equivalence checking**, i.e. checking whether two states are equivalent, and (*ii*) **search node expansion**, i.e. identifying the next AND-OR arcs. The experimental evaluation of De Giacomo *et al.*'s technique is rather impressive, as its implementation Cynthia, outperformed other state-of-the-art tools on challenging benchmarks, e.g. on the Nim benchmark [Bouton, 1901]. However, as already acknowledged by the authors (cfr. Section 5 of [De Giacomo *et al.*, 2022]), the tool performed poorly on the variant of the *Double Counters* benchmark used in [De Giacomo *et al.*, 2022]. I discovered that the main reason is that the search gets stuck with the search node expansion to compute the next agent's and environment's moves, whose number grows exponentially with the scaling parameter n, the number of bits of the counters. In general, I identify at least three factors that hinder the scalability of Cynthia:

(*i*) Disjoint & covering moves. Ltlfsyn, which naively enumerates all the exponentially-many agent's and env's moves, has been surpassed by Cynthia. Cynthia is able to branch on disjoint and covering propositional formulas, rather than individual evaluations of agent's and env's variables, and therefore ending up, most of the times, in a more succinct representation of the next players' moves. Nevertheless, for problem instances where the branching factor is very high, the compilation by means of SDDs does not bring much more benefits than exhaustive enumeration, ending up in a huge computational overhead with little usefulness.

(*ii*) No visit before *all* moves are computed. The search algorithm is constrained by how the identification of the next moves works. That is, the search procedure cannot visit children nodes before all OR arcs, and subsequent AND arcs, have been computed from the current OR-node being expanded. Obviously, a breadth-first search procedure (e.g. AO* [J. Nilsson, 1982]) will need to consider all the children of the current search node before proceeding. The point is that, if a search procedure does not need to know in advance all the children of the current node, like Algorithm 1, then it must be able to do so.

(*iii*) Monolithic. It is not necessary to tighten together the two tasks of state-equivalence checking and search node expansion. They can be implemented in different ways according to the desired computation trade-offs (e.g. space vs time).

3.1 A new framework.

My aim is to propose a new framework that tries to overcome the above limitations that we consider crucial for a scalable approach. To do so, I consider a slightly more general version of Algorithm 1. The generalization is not on the search algorithm being used, but rather on the building blocks that make any AND-OR search algorithm actually suitable for solving LTL_f synthesis in a forward fashion. In particular, I make a step further from the framework introduced in [De Giacomo *et al.*, 2022], which formalizes the search algorithm on top of the EXPAND function. Instead, the two primitive operations that I consider are: EQUIVALENCECHECK(n_1, n_2), that checks whether the search nodes n_1 and n_2 can be considered equivalent wrt the current AND-OR search problem; and GETARCS(n), that returns an *iterator* of AND-arcs (OR-arcs resp.) of the AND-node (OR-node, resp.) n. In Algorithm 1, the EQUIVALENCECHECK procedure is (implicitly) used to check whether a node has been already visited (e.g. see the INPATH function of Line 13) or to retrieve search tags (e.g. see ISSUCCESSNODE, ISFAILURENODE and ISTAGLOOP). The GETARCS procedure would be used in place of GETORARCS and GETANDARCS in Algorithm 1, at lines 21 and 22, respectively. For the rest of the paper, I consider such **modified Algorithm 1 as the basis of my techniques.**

The crucial observation is that GETARCS(n) does not require that the arcs of search node n have already been computed or, in other words, that the node n has been fully expanded (as done by EXPAND function). As per specification, GETARCS(n) is an iterator over the available moves from n. The concept of iterator is well-known in the computer science community as a way to decouple algorithms from containers [Gamma *et al.*, 1995]. More interestingly, a special case of iterators, *generators* [Murer *et al.*, 1996], would allow to compute the next players' moves iteratively "on-demand", therefore allowing a depth-first search algorithm to visit the next arc returned by the generator even if all arcs have not been computed yet. I will use a generator-based realization of the abstract function GETARCS in the next sections.

In fact, De Giacomo *et al.*'s approach can be seen as a special case of the proposed framework, in which both EQUIVALENCECHECK and GETARCS are implemented using SDDs: two search nodes are equivalent if they point to the same SDD node, and GETARCS is an iterator that simply scans the children of the root SDD node of n. However, this framework can easily overcome the limiting factors identified earlier in this section, namely: (*i*) computed moves do not have to be disjoint and covering (i.e. different moves that lead to the same successor are allowed, although preferably avoided); (*ii*) if GETARCS is implemented using a generator-like approach, the visit of a child node can happen far before the computation of all the available moves; and (*iii*) the two main search subtasks, state-equivalence checking and a search node expansion, are implemented by two potentially decoupled functions (EQUIVALENCECHECK and GETARCS, respectively).

4 DPLL-based Forward Synthesis

In this section, I describe my main novel approach for forward LTL_f synthesis of an LTL_f formula φ, as an instantiation of the abstract framework presented in Section 3. In particular, EQUIVALENCECHECK is implemented using BDDs, and GETARCS is implemented using a Davis-Putnam-Logemann-Loveland-like (DPLL) procedure (Algorithm 2). While the knowledge-compilation-based equivalence check is not new, as it is very similar to what has been already done for other forward LTL_f synthesis approaches, I claim the DPLL-based

GETARCS to be novel and effective for solving our problem, and it is one of the core contributions of the paper.

BDD-based EQUIVALENCECHECK. The BDD-based equivalence check is similar to the SDD-based equivalence check performed by Cynthia. That is, for a search node n, we take its associated LTL_f formula ψ with FORMULAOFNODE (remember that search node is associated to an LTL_f formula). Then, we compute $\mathsf{xnf}(\psi)$, which is propositionally equivalent to ψ. $\mathsf{xnf}(\psi)$, by construction, is defined over the set of variables $\mathcal{Y} \cup \mathcal{X} \cup \mathcal{Z}$, where $\mathcal{Z} = \bigcup_{\theta \in \mathsf{cl}(\varphi)} \{z_\alpha | \alpha \in \mathsf{pa}(\mathsf{xnf}(\theta)), \alpha \text{ not literal}\}$. Finally, we get its BDD representation, i.e. $B_\psi := \text{BDDREPRESENTATION}(\mathsf{xnf}(\psi)^p)$. We do these operations both for n_1 and n_2, yielding $B_{\mathsf{xnf}(\psi_1)}$ and $B_{\mathsf{xnf}(\psi_2)}$. The equivalence check whether the two BDDs point to the same BDD node ($B_{\mathsf{xnf}(\psi_1)} = B_{\mathsf{xnf}(\psi_2)}$). If that is the case then it means, thanks to the canonicity property of BDDs, that the associated (propositionalized) formulas are propositionally equivalent. I preferred the use of BDDs instead of SDDs since we do not need the decomposing feature of SDDs, and also because robust and optimized implementations for BDDs already exists, e.g. CUDD [Somenzi, 2016], with useful features such as dynamic variable reordering.

DPLL-based GETARCS. The DPLL algorithm [Davis and Putnam, 1960; Davis *et al.*, 1962] is a very famous algorithm for deciding the satisfiability of proposition logic formulas in conjunctive normal form (CNF). Many variants of it have been proposed that work for general non-clausal formulas [Thiffault *et al.*, 2004; Jain and Clarke, 2009], motivated by the fact that, quite often, conversion of a boolean formula to CNF is both unnecessary and undesirable, e.g. because of loss of structural information and due to the worst-case exponential blow-up of the size of the formula. I agree with this view, and in the following we assume to deal with propositionalized LTL_f formulas in non-clausal form.

I am interested in designing a DPLL-like procedure to identify the next moves and successor nodes from a search node n. My proposed procedure (Algorithm 2), like any DPLL procedure, runs by choosing a literal, assigning a truth value to it, simplifying the formula and then recursively applying the same procedure to the simplified formula, until there are no agent or environment variables to branch on. Both the computed set of assignments resulting from the sequence of recursive calls, ass (initialized at Line 3), and what remains of the formula $\phi = \mathsf{xnf}(\text{FORMULAOFNODE}(n))^p$ after the chosen literals have been replaced with their assigned truth value, are *yielded* such that they can be consumed by the caller function (see Line 17 and 28; the instruction **yield** allows a generator to provide a value to the caller).

Given a search node n, DPLLGETARCS returns a generator over pairs (move, node), where move is a mapping from variables to truth values (the absence of a variable is considered a *don't care*), and node is a LTL_f formula that, as required by mine and De Giacomo *et al.*'s search framework, represents a search node (either AND or OR). Depending on whether n is an OR-node or an AND-node, the DPLLGETORARCS function (Line 5) or the DPLLGETANDARCS function (Line 7) is called, respectively. The DPLLGETORARCS function takes in input a propositionalization of ψ, ϕ, and the current variables' assignment ass. If there is still some agent variable in $\mathcal{Y}$ to assign (Line 9), then we decide the next branching literal ℓ (by calling the function GETBRANCHINGLITERAL, Line 11), we substitute its truth value to the formula ϕ, and simplify it by calling the function REPLACE (Line 12), obtaining ϕ_ℓ. Then, we do the recursive call to DPLLGETORARCS with the new propositionalized formula ϕ_ℓ and updated assignment $ass \cup \{\ell\}$, and start generating the next moves with a fixed value for literal ℓ. Intuitively, this step represents a transition to another node of the search tree of a DPLL algorithm. The instruction **yield from** allows a generator to forward the generation of results to another generating function. When the generation terminates, the negated literal $\neg\ell$ is replaced to the original formula ϕ, yielding another propositionalized LTL_f formula $\phi_{\neg\ell}$, and the available moves starting from this branch are generated. Intuitively, the last step represents the exploration of the opposite branch of the current node of the DPLL search tree, with the branching literal ℓ set at the opposite truth value $\neg\ell$. Note that in the base case, we return the pair $(ass, \phi^{\mathsf{tf}})$, where ass contains all the chosen literals in the current final assignment, and ϕ^{tf} is the LTL_f formula that represents the next AND node. The DPLLGETANDARCS is analogous to DPLLGETORARCS but for AND nodes; therefore, it aims at finding an assignment of env variables $\mathcal{X}$ rather than of agent variables $\mathcal{Y}$. Another difference with DPLLGETANDARCS is that in the base case, we use the propositional formula Ψ (the result of the substitutions of chosen literals and the subsequent simplifications) to compute the next search node formula ψ', using the function RMNEXT, at Line 27. Note that, at this stage, Ψ is a propositional formula over $\mathcal{Z}$ state variables only. By Proposition 8, since $\Psi = \mathsf{xnf}(\psi)^p|_\sigma$, we have that $\psi' = \text{RMNEXT}(\Psi) = \mathsf{fp}(\psi, \sigma)$, i.e. the correct next state.

According to the needs of the search algorithm, the procedure can be run exhaustively, i.e. until all available moves from node n have been produced. Still, the simplification step can possibly avoid a large part of the naive search space over $\mathcal{Y}$ and $\mathcal{X}$; this is an improvement wrt the Ltlfsyn approach, which blindly enumerates all possible assignments. The simplification step recursively applies the usual propositional simplification rules, e.g. considering the absorbing or neutral boolean values of binary operators. I suggest to simplify the propositional formula to a great extent, but without resorting to any compilations. Instead, we leave the formula in non-clausal form, aiming at eliminating branching variables from the resulting formula. Such variables will be considered as *don't care* in the current assignment.

I argue that such kind of procedures, like the one described in Algorithm 2, are suitable for our use-case because of their depth-first nature, which implies a low-space requirement, and because of their "responsive" nature: a candidate move is proposed in linear time on the number of variables (possibly better thanks to simplifications). Note that Alg. 2 is an abstract specification that can be customized by different realizations of GETBRANCHINGLITERAL and REPLACE.

Theorem 2. *Modified Algorithm 1 with* BDDBASEDEQCHECK *for state-equivalence checking and Algorithm 2 for search node expansion is correct and always terminates.*

Proof sketch. Termination follows from canonicity of BDD

Algorithm 2 DPLL-based GETARCS

1: **function** DPLLGETARCS(n) **return** Gen[move, node]
2: $\psi \leftarrow \mathsf{xnf}(\text{FORMULAOFNODE}(n))$
3: $ass \leftarrow \{\}$ ▷ propositional assignment
4: **if** ISORNODE(n) **then**
5: **yield from** DPLLGETORARCS (ψ^p, ass)
6: **else**
7: **yield from** DPLLGETANDARCS (ψ^p, ass)
8: **function** DPLLGETORARCS(ϕ, ass)
9: $\mathcal{Y}' \leftarrow$ GETAGENTVARS(ϕ)
10: **if** $\mathcal{Y}' \neq \emptyset$ **then**
11: $\ell \leftarrow$ GETBRANCHINGLITERAL(ϕ)
12: $\phi_\ell \leftarrow$ REPLACE(ϕ, ℓ)
13: **yield from** DPLLGETORARCS($\phi_\ell, ass \cup \{\ell\}$)
14: $\phi_{\neg\ell} \leftarrow$ REPLACE($\phi, \neg\ell$)
15: **yield from** DPLLGETORARCS($\phi_{\neg\ell}, ass \cup \{\neg\ell\}$)
16: **else** ▷ No branching on agent variables available
17: **yield** $(ass, \phi^{\mathsf{tf}})$ ▷ ϕ^{tf} is the next AND node
18: **function** DPLLGETANDARCS(Ψ, ass)
19: $\mathcal{X}' \leftarrow$ GETENVVARS(Ψ)
20: **if** $\mathcal{X}' \neq \emptyset$ **then**
21: $\ell \leftarrow$ GETBRANCHINGLITERAL(Ψ)
22: $\Psi_\ell \leftarrow$ REPLACE(Ψ, ℓ)
23: **yield from** DPLLGETANDARCS($\Psi_\ell, ass \cup \ell$)
24: $\Psi_{\neg\ell} \leftarrow$ REPLACE($\Psi, \neg\ell$)
25: **yield from** DPLLGETANDARCS($\Psi_{\neg\ell}, ass \cup \neg\ell$)
26: **else** ▷ No branching on env variables available
27: $\psi' \leftarrow$ RMNEXT(Ψ)
28: **yield** (ass, ψ') ▷ ψ' is the next OR node

representation of search nodes and Theorem 4 of [De Giacomo *et al.*, 2022]. Correctness holds by observing that, by construction, Ψ at Line 27 is equal to $\mathsf{xnf}(\psi)^p|_\sigma$; putting it together with Proposition 8 and Theorem 5 of [De Giacomo *et al.*, 2022], we get the thesis. □

5 Hash-Consing-based Equivalence Check

In this section, I devise a variant of the search algorithm proposed in Section 4, where we replace the BDDBASEDEQCHECK with a check based on *structural equivalence*: two search nodes n_1 and n_2 are considered equivalent if their formulas ψ_1 and ψ_2 have the same syntax tree, i.e.: HASHCONSINGEQCHECK(n_1, n_2) := FORMULAOFNODE(n_1) = FORMULAOFNODE(n_2). To make the comparison fast, we can use *hash consing* [Deutsch, 1973] which is a technique used to share values that are structurally equal. Using hash consing, two formulas can be stated as structurally equivalent if they point to the same memory address, achieving constant time equality check. Unfortunately, we have the following negative result:

Theorem 3. *The modified Algorithm 1 with* HASHCONSINGEQCHECK *for* EQUIVALENCECHECK *and Algorithm 2 for* GETARCS *is sound but not complete for* LTL$_f$ *synthesis.*

Proof sketch. Soundness follows from correctness of DPLLGETARCS and by the soundness of hash-consing based equivalence check. To disprove completeness, consider the synthesis problem with $\varphi = \Box a \, \mathcal{U} \Diamond b$, $\mathcal{Y} = \{a\}$ and $\mathcal{X} = \{b\}$. Repeatedly taking agent-environment move corresponding to assignment $\sigma = \{a\}$ produces ever bigger, but propositionally equivalent, state formulas, ending up in an infinite recursion. See the supplementary material for more details. □

At the core of the issue is that, by how the formula progression works, there are some cases in which a new structurally different formula can be always produced by some particular sequence of applications of formula progression rules, although propositionally equivalent formulas have been already produced earlier during the search. Nevertheless, the hash-consing based equivalence check is very computationally cheap and, as we shall see in the experimental section, often it performs better than the BDD-based equivalence check.

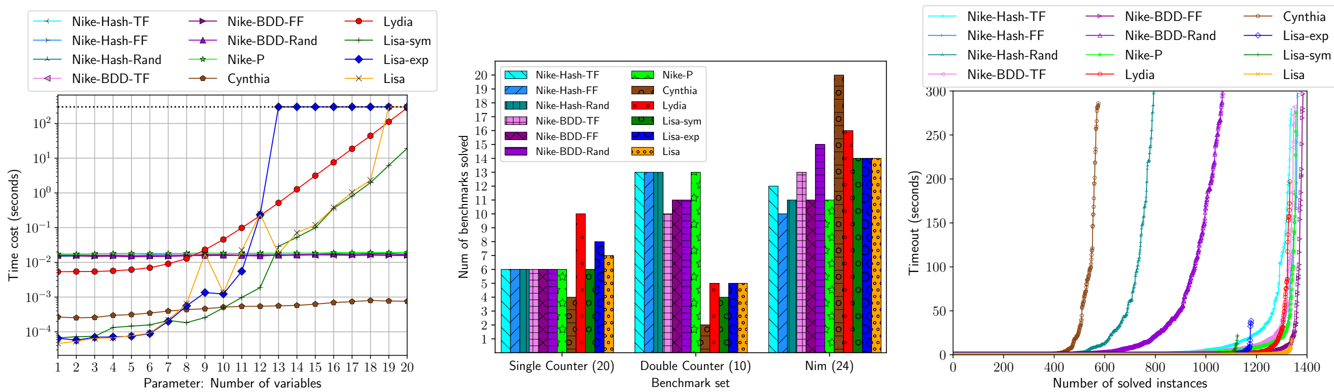
To guarantee the termination of this version of the search algorithm, I propose the following procedure: given a synthesis problem, first execute the modified Algorithm 1 with HASHCONSINGEQCHECK as equivalence check and DPLLGETARCS for search node expansion. As soon as, during the execution, the size of the formula of any generated search node becomes greater than a given threshold t, then abort the execution and resort to the search algorithm described in Section 4, i.e. Algorithm 1 based on BDDBASEDEQCHECK and DPLLGETARCS. In other words, if the problem does not present the pathological corner case shown in the proof of Theorem 4, then try to solve it, without getting stuck with onerous BDD-based compilations. The threshold guarantees that only a finite number of structurally equivalent formulas can be computed. Empirically, we found that a good threshold that suitably postpones the detection of pathological instances is three times the size of the initial formula: $t = 3 \cdot |\varphi|$.

6 Implementation and Experiments

I implemented the presented synthesis methods in a tool called Nike, in C++ [1]. Nike takes in input an LTL$_f$ synthesis problem and constructs a strategy that realizes the specification, if one exists. I use the CUDD library (github.com/ivmai/cudd) to handle all BDD related operations. Nike, as Cynthia and Ltlfsyn, applies some optimizations to speed up the synthesis procedure. First, when visiting an OR-node n for the first time, we perform the preprocessing techniques described in [Xiao *et al.*, 2021]. More specifically, we check: (*i*) there exists a one-step strategy that reaches accepting states from n, then n is tagged as success; or (*ii*) there does not exist an agent move that can avoid sink state (a non-accepting state only going back to itself) from n, then n is tagged as failure.

Nike can run in two modes: using BDD-based state-eq checking (BDD), and hash-consing-based state eq checking (Hash). In the DPLL-based search node expansion, I considered variables in alphabetical order, and I combined them with three simple branching strategies: *True-first* (TF) that first sets variables to true, *False-first* (FF) that first sets variables to false; and *Random* (Rand) that sets variables at random. This yields six combinations of Nike that I included in these experiments. I also include a parallel version of Nike, Nike-P, that runs in hash-consing-based modes all the three branching strategies in parallel.

[1] Source code will be available upon request.



(a) GF-pattern.

(b) *Two-player-Games*.

(c) *Random*.

Figure 1: Comparison results on all benchmarks.

Experimental Methodology. I evaluated the efficiency of all variants of Nike, by comparing against the following tools: Lisa [Bansal *et al.*, 2020a] and Lydia [De Giacomo and Favorito, 2021] are state-of-the-art backward LTL_f synthesis approaches. Both tools compute the complete DFA first, and then solve an adversarial reachability game following the symbolic backward computation technique described in [Zhu *et al.*, 2017b]. I excluded Ltlfsyn from the comparison because it has been already shown to be superseded by Cynthia.

Experiment Setup. Experiments were run on a VM instance on Google Cloud, type `c2-standard-4`, endowed with Intel(R) Xeon(R) CPU 3.10GHz, 4 logical CPU threads, 16 GB of memory and 300 seconds of time limit. The correctness of Nike was empirically verified by comparing the results with those from all baseline tools. No inconsistency was found.

Benchmarks. We collected, in total, 1494 LTL_f synthesis instances from literature: 40 Patterns instances (*GF*- and *U*-patterns) [Xiao *et al.*, 2021]; 54 Two-player-Games instances: *Single-Counter*, *Double-Counter* and *Nim* [Tabajara and Vardi, 2019; Bansal *et al.*, 2020a]; and 1400 Random instances [Zhu *et al.*, 2017b; De Giacomo and Favorito, 2021].

Analysis. Figure 4 shows the running time of each tool on every instance of the *GF*-pattern dataset. Across these instances, we can observe that all variants of Nike solve instances very quickly, thanks to the pre-processing techniques. This is done with much less time comparing to backward approaches, represented by Lisa and Lydia, simply because these tool do not have such optimizations. Cynthia solved in less time in logarithmic scale, but I attribute this to the set up time of the CUDD BDD manager that worsen the performances. Nevertheless, this amount to a negligible time cost difference of $\ll 1$ second. Similar results are for the *U*-pattern dataset, shown in the supplementary material. On the *Two-player-Games* benchmarks, see Figure 5, we observe that Nike variants dominate all other tools on the *Double-Counter* instances, while competing with backward approaches on the other instances. On *Nim*, Cynthia is the best performing tool, but on the other benchmarks Nike shows to be better. The Nike-BDD combinations performs slightly worse on Double Counter than the Nike-Hash combinations. On the *Random* benchmarks, all variants of Nike, but the ones using Rand branching strategy, are competitive with state-of-the-art backward approaches, and far better than Cynthia.

It is clear from the plots that Nike, in general, shows an overall better performance than Cynthia, illustrating the efficiency and better scalability of my approach. In particular, there is a notable outperformance of Cynthia on the *Double-Counter* and in the *Random* instances. I attribute this to the ability of Nike to not being stuck with compilation processes that can easily become intractable, both on hand-designed datasets like *Double-Counter*, and in randomly generated intractable cases. Moreover, despite the simplicity of the DPLL-based expansion, performances are very surprising with respect to backward approaches; this suggests that my approach is very promising and worth of future research. The worse performance of the Rand branching strategy on the *Random* benchmark can be explained by the fact that the TF and the FF strategies might exploit a particular problem structure of these instances, that allow to easily arrive to success nodes or failure nodes, and saves the algorithm to explore more moves thanks to the short-circuit evaluation of the search outcome (see Lines 24 and 25 in the modified Algorithm 1). The best configuration is Nike-BDD-FF, which suggests that for this benchmark the state compilation is not too hard and the canonicity of the representation helps to prevent the revisit of propositionally-equivalent states.

7 Conclusions

I proposed the best forward search LTL_f synthesis approach so far, and the first that is truly competitive with the considered state-of-the-art tools based on backward computation (as in the *Random* benchmark). I think this work sets the foundations for a new family of forward LTL_f synthesis algorithms, and opens several research avenues for investigating effective branching heuristics [Silva, 1999] for the DPLL-based search graph expansion (e.g. non-chronological backtracking), or better termination strategies for searching with hash-consing-based state-equivalence checking.

References

[Bacchus and Kabanza, 1998] F. Bacchus and F. Kabanza. Planning for temporally extended goals. *Ann. Math. Artif. Intell.*, 22(1-2), 1998.

[Baier and Katoen, 2008] C. Baier and JP. Katoen. *Principles of model checking*. 2008.

[Bansal *et al.*, 2020a] S. Bansal, Y. Li, L. M. Tabajara, and M. Y. Vardi. Hybrid compositional reasoning for reactive synthesis from finite-horizon specifications. In *AAAI*, 2020.

[Bansal *et al.*, 2020b] S. Bansal, Y. Li, L. M. Tabajara, and M. Y. Vardi. Hybrid Compositional Reasoning for Reactive Synthesis from Finite-Horizon Specifications. In *AAAI*, 2020.

[Bertoli *et al.*, 2006] P. Bertoli, A. Cimatti, M. Roveri, and P. Traverso. Strong planning under partial observability. *Artif. Intell.*, 170(4-5), 2006.

[Bouton, 1901] C. L. Bouton. Nim, a game with a complete mathematical theory. *Annals of Mathematics*, 3, 1901.

[Brafman *et al.*, 2018] R. I. Brafman, G. De Giacomo, and F. Patrizi. LTL_f/LDL_f non-markovian rewards. In *AAAI*, 2018.

[Bryant, 1992] R. E. Bryant. Symbolic Boolean Manipulation with Ordered Binary-Decision Diagrams. *ACM Comput. Surv.*, 24(3), 1992.

[Chakrabarti, 1994] P. P. Chakrabarti. Algorithms for searching explicit AND/OR graphs and their applications to problem reduction search. *Artif. Intell.*, 65(2), 1994.

[Church, 1963] Alonzo Church. Application of recursive arithmetic to the problem of circuit synthesis. *Journal of Symbolic Logic*, 28(4), 1963.

[Cimatti *et al.*, 1998] A. Cimatti, M. Roveri, and P. Traverso. Strong planning in non-deterministic domains via model checking. In *AIPS*, 1998.

[Cimatti *et al.*, 2003] A. Cimatti, M. Pistore, M. Roveri, and P. Traverso. Weak, strong, and strong cyclic planning via symbolic model checking. 1–2(147), 2003.

[Darwiche and Marquis, 2002] A. Darwiche and P. Marquis. A knowledge compilation map. *J. Artif. Intell. Res.*, 17:229–264, 2002.

[Darwiche, 2011] A. Darwiche. SDD: A new canonical representation of propositional knowledge bases. In *IJCAI*, 2011.

[Davis and Putnam, 1960] Martin Davis and Hilary Putnam. A computing procedure for quantification theory. *J. ACM*, 7(3):201–215, 1960.

[Davis *et al.*, 1962] Martin Davis, George Logemann, and Donald W. Loveland. A machine program for theorem-proving. *Commun. ACM*, 5(7):394–397, 1962.

[De Giacomo and Favorito, 2021] G. De Giacomo and M. Favorito. Compositional approach to translate LTL_f/LDL_f into deterministic finite automata. In *ICAPS*, 2021.

[De Giacomo and Vardi, 2013] G. De Giacomo and M. Y. Vardi. Linear Temporal Logic and Linear Dynamic Logic on Finite Traces. In *IJCAI*, 2013.

[De Giacomo and Vardi, 2015] G. De Giacomo and M. Y. Vardi. Synthesis for LTL and LDL on Finite Traces. In *IJCAI*, 2015.

[De Giacomo *et al.*, 2022] Giuseppe De Giacomo, Marco Favorito, Jianwen Li, Moshe Y. Vardi, Shengping Xiao, and Shufang Zhu. Ltlf synthesis as AND-OR graph search: Knowledge compilation at work. In *IJCAI*, pages 2591–2598. ijcai.org, 2022.

[Deutsch, 1973] L Peter Deutsch. An interactive program verifier. 1973.

[Ehlers, 2010] R. Ehlers. Symbolic Bounded Synthesis. In *CAV*, 2010.

[Emerson, 1990] E. Allen Emerson. Temporal and modal logic. In *Handbook of Theoretical Computer Science*, 1990.

[Gamma *et al.*, 1995] Erich Gamma, Ralph Johnson, Richard Helm, Ralph E Johnson, and John Vlissides. *Design patterns: elements of reusable object-oriented software*. Pearson Deutschland GmbH, 1995.

[Geffner and Bonet, 2013] H. Geffner and B. Bonet. *A Concise Introduction to Models and Methods for Automated Planning*. 2013.

[Ghallab *et al.*, 2004] M. Ghallab, D. S. Nau, and P. Traverso. *Automated planning - theory and practice*. 2004.

[Goldman and Boddy, 1996] R. P. Goldman and M. S. Boddy. Expressive planning and explicit knowledge. In *AIPS*, 1996.

[Haslum *et al.*, 2019] P. Haslum, N. Lipovetzky, D. Magazzeni, and C. Muise. *An Introduction to the Planning Domain Definition Language*. 2019.

[J. Nilsson, 1982] N. J. Nilsson. *Principles of Artificial Intelligence*. 1982.

[Jain and Clarke, 2009] Himanshu Jain and Edmund M. Clarke. Efficient SAT solving for non-clausal formulas using dpll, graphs, and watched cuts. In *DAC*, pages 563–568. ACM, 2009.

[Jiménez and Torras, 2000] P. Jiménez and C. Torras. An efficient algorithm for searching implicit AND/OR graphs with cycles. *Artif. Intell.*, 124(1), 2000.

[Jobstmann and Bloem, 2006] B. Jobstmann and R. Bloem. Optimizations for LTL synthesis. In *FMCAD*, 2006.

[Li *et al.*, 2019] J. Li, K. Y. Rozier, G. Pu, Y. Zhang, and M. Y. Vardi. Sat-based explicit LTL_f satisfiability checking. In *AAAI*, 2019.

[Mahanti and Bagchi, 1985] A. Mahanti and A. Bagchi. AND/OR graph heuristic search methods. *J. ACM*, 32(1), 1985.

[Mattmüller *et al.*, 2010] R. Mattmüller, M. Ortlieb, M. Helmert, and P. Bercher. Pattern database heuristics for fully observable nondeterministic planning. In *ICAPS*, 2010.

[Mattmüller, 2013] R. Mattmüller. *Informed progression search for fully observable nondeterministic planning*. PhD thesis, 2013.

[Murer *et al.*, 1996] Stephan Murer, Stephen Omohundro, David Stoutamire, and Clemens Szyperski. Iteration abstraction in sather. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 18(1):1–15, 1996.

[Nilsson, 1971] N. J. Nilsson. *Problem-solving methods in artificial intelligence*. 1971.

[Pnueli and Rosner, 1989] A. Pnueli and R. Rosner. On the Synthesis of a Reactive Module. In *POPL*, 1989.

[Pnueli, 1977] A. Pnueli. The temporal logic of programs. In *FOCS*, 1977.

[Reif, 1984] J. H. Reif. The complexity of two-player games of incomplete information. *JCSS*, 29(2), 1984.

[Rintanen, 2004] J. Rintanen. Complexity of planning with partial observability. In *ICAPS*, 2004.

[Scutellà, 1990] M. G. Scutellà. A note on dowling and gallier's top-down algorithm for propositional horn satisfiability. *J. Log. Program.*, 8(3):265–273, 1990.

[Silva, 1999] João P. Marques Silva. The impact of branching heuristics in propositional satisfiability algorithms. In *EPIA*, volume 1695 of *Lecture Notes in Computer Science*, pages 62–74. Springer, 1999.

[Somenzi, 2016] Fabio Somenzi. CUDD: CU Decision Diagram Package 3.0.0. Universiy of Colorado at Boulder. 2016.

[Tabajara and Vardi, 2019] L. M. Tabajara and M. Y. Vardi. Partitioning Techniques in LTL_f Synthesis. In *IJCAI*, 2019.

[Thanh To *et al.*, 2009] S. Thanh To, E. Pontelli, and T. Cao Son. A conformant planner with explicit disjunctive representation of belief states. In *ICAPS*, 2009.

[Thiffault *et al.*, 2004] Christian Thiffault, Fahiem Bacchus, and Toby Walsh. Solving non-clausal formulas with DPLL search. In *SAT*, 2004.

[Xiao *et al.*, 2021] S. Xiao, J. Li, S. Zhu, Y. Shi, G. Pu, and M. Y. Vardi. On-the-fly synthesis for LTL over finite traces. In *AAAI*, 2021.

[Zhu *et al.*, 2017a] S. Zhu, L. M. Tabajara, J. Li, G. Pu, and M. Y. Vardi. Symbolic LTL_f Synthesis. In *IJCAI*, 2017.

[Zhu *et al.*, 2017b] Shufang Zhu, Lucas M. Tabajara, Jianwen Li, Geguang Pu, and Moshe Y. Vardi. A Symbolic Approach to Safety LTL Synthesis. In *HVC*, 2017.

A Preliminaries

LTL_f **Basics.** Linear Temporal Logic over finite traces, called LTL_f [De Giacomo and Vardi, 2013] is a variant of Linear Temporal Logic (LTL) [Baier and Katoen, 2008] that is interpreted over finite traces rather than infinite traces (as in LTL). Given a set of propositions $\mathcal{P}$, the syntax of LTL_f is identical to LTL, and defined as (wlog, we require LTL_f formulas are in Negation Normal Form (NNF), i.e., negations only occur in front of atomic propositions): $\varphi ::= tt \mid ff \mid p \mid \neg p \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \circ\varphi \mid \bullet\varphi \mid \varphi_1 \mathcal{U} \varphi_2 \mid \varphi_1 \mathcal{R} \varphi_2$. tt is always true, ff is always false; $p \in \mathcal{P}$ is an *atom*, and $\neg p$ is a *negated atom* (a literal l is an atom or the negation of an atom); $\wedge$ (And) and $\vee$ (Or) are the Boolean connectives; and $\circ$ (Next), $\bullet$ (Weak Next), $\mathcal{U}$ (Until) and $\mathcal{R}$ (Release) are temporal connectives. We use the usual abbreviations $true \equiv p \vee \neg p$, $false \equiv p \wedge \neg p$, $\Diamond\varphi \equiv true\, \mathcal{U}\, \varphi$ and $\Box\varphi \equiv false\, \mathcal{R}\, \varphi$. Also for convenience we consider traces $\rho \in (2^{\mathcal{P}})^*$, i.e., we consider also empty traces ϵ as in [Brafman *et al.*, 2018]. More specifically, a trace $\rho = \rho[0], \rho[1], \ldots \in (2^{\mathcal{P}})^*$ is a finite sequence, where $\rho[i]$ $(0 \leq i < |\rho|)$ denotes the i-th interpretation of ρ, which can be considered as the set of propositions that are $true$ at instant i, and $|\rho|$ represents the length of ρ. We have that $\epsilon \models \varphi$ if φ is tt, an $\mathcal{R}$-formula or $\bullet$-formula, hence $\epsilon \models \Box false$. $\epsilon \not\models \varphi$ if φ is ff, a literal, $\mathcal{U}$-formula or $\circ$-formula, hence $\epsilon \not\models \Diamond true$. We consider the semantics of LTL_f as presented in [Brafman *et al.*, 2018].

We denote by $\mathsf{cl}(\varphi)$ the set of subformulas of φ, including tt and ff. We denote by $\mathsf{pa}(\varphi) \subseteq \mathsf{cl}(\varphi)$ the set of literals and temporal subformulas of φ whose primary connective is temporal [Li *et al.*, 2019]. Formally, for an LTL_f formula φ in NNF, we have $\mathsf{pa}(\varphi) = \{\varphi\}$ if φ is a literal or temporal formula; and $\mathsf{pa}(\varphi) = \mathsf{pa}(\varphi_1) \cup \mathsf{pa}(\varphi_2)$ if $\varphi = (\varphi_1 \wedge \varphi_2)$ or $\varphi = (\varphi_1 \vee \varphi_2)$. Having LTL_f formula φ, replacing every temporal formula $\psi \in \mathsf{pa}(\varphi)$ with a propositional variable a_ψ gives us a propositional formula φ^p; we call this operation *propositionalization of* φ. Note that $\varphi^p \in \mathcal{B}^+(\mathsf{cl}(\varphi))$, i.e. φ^p is a positive Boolean formula over variables $\mathsf{cl}(\varphi)$. Let $\phi = \varphi^p$, we denote with $\phi^{\mathsf{tf}} = \varphi$ the inverse operation of $\cdot^p$. Two formulas φ_1 and φ_2 are propositionally equivalent, denoted by $\varphi_1 \sim_p \varphi_2$, if, $C \models \varphi_1^p \leftrightarrow C \models \varphi_2^p$ holds for every propositional assignment $C \in 2^{\mathsf{pa}(\varphi_1) \cup \mathsf{pa}(\varphi_2)}$.

An LTL_f formula φ is in neXt Normal Form (XNF) if $\mathsf{pa}(\varphi)$ only includes literals, $\circ$- and $\bullet$-formulas. For an LTL_f formula φ in NNF, we can obtain its XNF by transformation function $\mathsf{xnf}(\varphi)$, defined as follows:

- $\mathsf{xnf}(\varphi) = \varphi$ if φ is a literal, $\Box false$, $\Diamond true$, $\circ$-, $\bullet$-formula;
- $\mathsf{xnf}(\varphi_1 \wedge \varphi_2) = \mathsf{xnf}(\varphi_1) \wedge \mathsf{xnf}(\varphi_2)$;
- $\mathsf{xnf}(\varphi_1 \vee \varphi_2) = \mathsf{xnf}(\varphi_1) \vee \mathsf{xnf}(\varphi_2)$;
- $\mathsf{xnf}(\varphi_1 \mathcal{U} \varphi_2) = (\mathsf{xnf}(\varphi_2) \wedge \Diamond true) \vee (\mathsf{xnf}(\varphi_1) \wedge \circ(\varphi_1 \mathcal{U} \varphi_2))$;
- $\mathsf{xnf}(\varphi_1 \mathcal{R} \varphi_2) = (\mathsf{xnf}(\varphi_2) \vee \Box false) \wedge (\mathsf{xnf}(\varphi_1) \vee \bullet(\varphi_1 \mathcal{R} \varphi_2))$.

Note that $\Diamond true$ (resp. $\Box false$) guarantees that empty trace is not (resp. is) accepted by $\mathcal{U}$-formulas (resp. $\mathcal{R}$-formulas).

Theorem 4 ([Li *et al.*, 2019])**.** *Every* LTL_f *formula* φ *in* NNF *can be converted, with linear time in the formula size, to an equivalent formula in* XNF*, denoted by* $\mathsf{xnf}(\varphi)$*.*

LTL_f **Formula Progression [De Giacomo *et al.*, 2022].** Consider an LTL_f formula φ over $\mathcal{P}$ and a finite trace $\rho = \rho[0], \rho[1], \ldots \in (2^{\mathcal{P}})^*$, in order to have $\rho \models \varphi$, we can start from φ, progress or push φ through ρ. The idea behind *formula progression* is to consider LTL_f formula φ into a requirement about *now* $\rho[i]$, which can be checked straightaway, and a requirement about the future that has to hold on the yet unavailable suffix. That is to say, formula progression looks at $\rho[i]$ and φ, and progresses a new formula $\mathsf{fp}(\varphi, \rho[i])$ such that $\rho, i \models \varphi$ iff $\rho, i+1 \models \mathsf{fp}(\varphi, \rho[i])$. This procedure is analogous to DFA reading trace ρ, where reaching accepting states is essentially achieved by taking one transition after another. Formula progression has been studied in prior work, cf. [Emerson, 1990; Bacchus and Kabanza, 1998]. Here we use the formalization provided in [De Giacomo *et al.*, 2022].

Note that, since ρ is a finite trace, it is necessary to clarify when the trace ends. To do so, two new formulas are introduced: $\Box false$ and $\Diamond true$, which, intuitively, refer to *finite trace ends* and *finite trace not ends*, respectively. For simplicity, we enrich $\mathsf{cl}(\varphi)$, the set of proper subformulas of φ, to include them such that $\mathsf{cl}(\varphi)$ is reloaded as $\mathsf{cl}(\varphi) \cup \mathsf{cl}(\Diamond true) \cup \mathsf{cl}(\Box false)$.

For an LTL_f formula φ in NNF, the *progression function* $\mathsf{fp}(\varphi, \sigma)$, where $\sigma \in 2^{\mathcal{P}}$, is defined as follows:

- $\mathsf{fp}(tt, \sigma) = tt$ and $\mathsf{fp}(ff, \sigma) = ff$;
- $\mathsf{fp}(p, \sigma) = tt$ if $p \in \sigma$, otherwise ff;
- $\mathsf{fp}(\neg p, \sigma) = tt$ if $p \notin \sigma$, otherwise ff;
- $\mathsf{fp}(\varphi_1 \wedge \varphi_2, \sigma) = \mathsf{fp}(\varphi_1, \sigma) \wedge \mathsf{fp}(\varphi_2, \sigma)$;
- $\mathsf{fp}(\varphi_1 \vee \varphi_2, \sigma) = \mathsf{fp}(\varphi_1, \sigma) \vee \mathsf{fp}(\varphi_2, \sigma)$;
- $\mathsf{fp}(\circ\varphi, \sigma) = \varphi \wedge \Diamond true$;
- $\mathsf{fp}(\bullet\varphi, \sigma) = \varphi \vee \Box false$;
- $\mathsf{fp}(\varphi_1 \mathcal{U} \varphi_2, \sigma) = \mathsf{fp}(\varphi_2, \sigma) \vee (\mathsf{fp}(\varphi_1, \sigma) \wedge \mathsf{fp}(\circ(\varphi_1 \mathcal{U} \varphi_2), \sigma))$;
- $\mathsf{fp}(\varphi_1 \mathcal{R} \varphi_2, \sigma) = \mathsf{fp}(\varphi_2, \sigma) \wedge (\mathsf{fp}(\varphi_1, \sigma) \vee \mathsf{fp}(\bullet(\varphi_1 \mathcal{R} \varphi_2), \sigma))$.

Note that $\mathsf{fp}(\varphi, \sigma)$ is a positive Boolean formula on $\mathsf{cl}(\varphi)$, i.e., $\mathsf{fp}(\varphi, \sigma) \in \mathcal{B}^+(\mathsf{cl}(\varphi))$. The following two propositions show that $\mathsf{fp}(\varphi, \sigma)$ strictly follows LTL_f semantics and retains the propositional behavior of LTL_f formulas.

Proposition 5. *Let* φ *be an* LTL_f *formula over* $\mathcal{P}$ *in* NNF*,* ρ *be a finite nonempty trace,* $\mathsf{fp}(\varphi, \sigma)$ *be as above. We have that* $\rho, i \models \varphi$ *iff* $\rho, i+1 \models \mathsf{fp}(\varphi, \rho[i])$*.*

Proposition 6. *Let* φ *and* ψ *be two* LTL_f *formulas over* $\mathcal{P}$ *in* NNF *s.t.* $\varphi \sim_p \psi$*, and* $\sigma \in 2^{\mathcal{P}}$*. Then* $\mathsf{fp}(\varphi, \sigma) \sim_p \mathsf{fp}(\psi, \sigma)$ *holds.*

We generalize LTL_f formula progression from single instants to finite traces by defining $\mathsf{fp}(\varphi, \epsilon) = \varphi$, and $\mathsf{fp}(\varphi, \sigma u) = \mathsf{fp}(\mathsf{fp}(\varphi, \sigma), u)$, where $\sigma \in 2^{\mathcal{P}}$ and $u \in (2^{\mathcal{P}})^*$.

Proposition 7. *Let* φ *be an* LTL_f *formula over* $\mathcal{P}$ *in* NNF*,* ρ *be a finite trace. We have that* $\rho \models \varphi$ *iff* $\epsilon \models \mathsf{fp}(\varphi, \rho)$*.*

We take the definition of the *remove-next* function RMNEXT from [De Giacomo *et al.*, 2022], defined over propositionalized LTL_f formulas in XNF, φ^p:

- $\text{RmNext}(\Diamond true) = tt$, $\text{RmNext}(\Box false) = ff$
- $\text{RmNext}(\varphi_1 \wedge \varphi_2) = \text{RmNext}(\varphi_1) \wedge \text{RmNext}(\varphi_2)$
- $\text{RmNext}(\varphi_1 \vee \varphi_2) = \text{RmNext}(\varphi_1) \vee \text{RmNext}(\varphi_2)$
- $\text{RmNext}(\circ\varphi) = \varphi \wedge \Diamond true$, $\text{RmNext}(\bullet\varphi) = \varphi \vee \Box false$

Note that RMNEXT applies to neither $\mathcal{U}$-,$\mathcal{R}$- formulas, since they do not appear in XNF, nor literals (p, $\neg p$), as the input of the function is a propositionalized LTL_f formula in XNF form. We have the following proposition:

Proposition 8. *Given an* LTL_f *formula* φ *in* NNF, $\forall\sigma \in 2^{\mathcal{P}}, \mathsf{fp}(\varphi,\sigma) \equiv \textsc{RmNext}(\mathsf{xnf}(\varphi)^p|_\sigma)$*, where* $\mathsf{xnf}(\varphi)^p|_\sigma$ *stands for evaluating* σ *on* $\mathsf{xnf}(\varphi)^p$.

AND-OR Graph Search. Being a popular topic in AI, AND-OR graph search has attracted extensive studies. Following [Nilsson, 1971; J. Nilsson, 1982], an AND/OR graph can be considered as a generalization of a directed graph, where there are a set of nodes $\mathcal{V}$ and generalized connectors (edges) between nodes. Every connector links one single node $v \in \mathcal{V}$ to a set of nodes $V \subseteq \mathcal{V}$, where n is the number of nodes in the graph. A connector is called an AND (resp. OR) connector, if there is a logical AND (resp. OR) relationship among V. It should be noted that in this work I only focus on specific AND-OR graphs, where every node has only one connector leading to its successor nodes. Therefore, we have AND-nodes with an AND connector, and OR-nodes with an OR connector. Moreover, the set of goal nodes V_g only consists of OR-nodes.

The AND-OR graph search problem was first introduced in [Nilsson, 1971]. Intuitively speaking, the searching procedure aims to find a winning plan that encodes a path leading from the initial node to goal nodes. It is possible to involve both kinds of nodes in the winning plan, therefore, the plan lists one outgoing option at OR-nodes, and all outgoing options at AND-nodes leading to branches. Therefore, a winning plan is essentially a tree such that all leaves are goal nodes. There has been extensive studies on AND-OR graph search techniques [Mahanti and Bagchi, 1985; Chakrabarti, 1994; Jiménez and Torras, 2000], and have been utilized in a lot of applications, e.g., FOND planning [Mattmüller *et al.*, 2010; Mattmüller, 2013; Geffner and Bonet, 2013].

Knowledge Compilation: BDDs and SDDs. Knowledge Compilation [Darwiche and Marquis, 2002] is a family of approaches for dealing with the computational intractability of general propositional reasoning. A propositional theory is compiled off-line into a target language, which is then used on-line to answer a large number of queries in polytime. The key motivation behind knowledge compilation is to push as much of the computational overhead into the off-line phase, which is amortized over all on-line queries. There are a plethora of knowledge compilation techniques. Perhaps the first knowledge compilation technique are (Ordered) Binary Decision Diagrams (BDDs) [Bryant, 1992], where in order to represent a Boolean function, the classical method is applying Shannon decomposition. Intuitively, BDD decomposes Boolean functions with one variable at a time. Therefore, the canonicity of BDD is determined wrt a specific ordering of variables. Crucially, propositional equivalence between two propositional formulas can be done in constant time once both formulas are converted into BDDs. The more recent Sentential Decision Diagrams (SDDs) [Darwiche, 2011] utilize a more general decomposition technique that decomposes Boolean functions with a set of variables at each round. Let $f(\mathcal{Y}\cup\mathcal{X})$ be a Boolean function over variables $\mathcal{Y}\cup\mathcal{X}$, where $\mathcal{Y},\mathcal{X}$ are disjoint. Given an $(\mathcal{Y},\mathcal{X})$-partition, where $\mathcal{Y}$ variables are considered *primary* and $\mathcal{X}$ variables are considered *subsequent*, the SDD of f, with respect to the $(\mathcal{Y},\mathcal{X})$-partition, can be written as $\bigvee_{i=1}^{n}[\mathsf{prime}_i(\mathcal{Y}) \wedge \mathsf{sub}_i(\mathcal{X})]$. Intuitively, SDD decomposes f into n children, each of which consists of Boolean functions $\mathsf{prime}_i(\mathcal{Y})$ (what are satisfied *in primary*) and $\mathsf{sub}_i(\mathcal{X})$ (what should be satisfied *in subsequent*, according to $\mathsf{prime}_i(\mathcal{Y})$). In particular, besides that all the primes are disjoint and covering, i.e., $\mathsf{prime}_i \wedge \mathsf{prime}_j = \mathit{false}$ for $i \neq j$, and $\bigvee_{i=1}^{n} \mathsf{prime}_i = true$, SDD also guarantees that all the subs are compressed, i.e., $\mathsf{sub}_i(\mathcal{X}) \neq \mathsf{sub}_j(\mathcal{X})$ for $i \neq j$. Hence, the canonicity of SDDs is determined wrt a specific partition of variables.

LTL_f **Synthesis** The problem of LTL_f synthesis is described as a tuple $(\varphi,\mathcal{X},\mathcal{Y})$, where φ is an LTL_f formula over $\mathcal{X}\cup\mathcal{Y}$, and $\mathcal{X},\mathcal{Y}$ are two disjoint sets of variables controlled by the *environment* and the *agent*, respectively.

Definition 2. *The* synthesis *problem* $(\varphi,\mathcal{X},\mathcal{Y})$ *aims to computing a strategy* $g : (2^{\mathcal{X}})^* \to 2^{\mathcal{Y}}$*, such that for an arbitrary infinite sequence* $\lambda = X_0, X_1, \ldots \in (2^{\mathcal{X}})^\omega$*, we can find* $k \geq 0$ *such that* $\rho^k \models \varphi$*, where* $\rho^k = (X_0 \cup g(\epsilon)), (X_1 \cup g(X_0)), \ldots, (X_k \cup g(X_0, X_1, \ldots, X_{k-1}))$*. If such a strategy does not exist, then* φ *is unrealizable.*

LTL_f synthesis can be solved by reducing to an adversarial reachability game on the corresponding Deterministic Finite Automaton (DFA) [De Giacomo and Vardi, 2015]. Hence, a strategy can also be represented as a finite-state controller $g : \mathcal{S} \mapsto 2^{\mathcal{Y}}$, where $\mathcal{S}$ denotes the state space of the DFA.

B Proofs

B.1 Algorithm 1 + BddBasedEqCheck + DpllGetArcs

I now formally argue about the correctness of Algorithm 1 when combined with BddBasedEqCheck (Algorithm 2) and DpllGetArcs (Algorithm 3).

Lemma 1. *Let* $(\varphi,\mathcal{X},\mathcal{Y})$ *be a* LTL_f *synthesis problem instance. The* BddBasedEqCheck *procedure for such instance induces a search space for Algorithm 1 with no more than* $2^{2^{|\mathcal{O}(\mathsf{cl}(\varphi))|}}$ *search nodes.*

Proof. Any LTL_f formula ψ associated to some search node n of Algorithm 1 is such that $\mathsf{xnf}(\psi)^p \in \mathcal{B}^+(\mathcal{Y}\cup\mathcal{X}\cup\mathcal{Z})$. Since there are at most $2^{|\mathcal{Y}\cup\mathcal{X}\cup\mathcal{Z}|}$ models, thanks to the canonicity property of BDDs, there can be at most $2^{2^{|\mathcal{Y}\cup\mathcal{X}\cup\mathcal{Z}|}}$ propositionally equivalent formulas. Since $\mathcal{Y}\cup\mathcal{X}\cup\mathcal{Z} = \mathcal{O}(\mathsf{cl}(\varphi))$, we get the claim. ☐

Lemma 2. *Let* $(\varphi,\mathcal{X},\mathcal{Y})$ *be a* LTL_f *synthesis problem instance. The* DpllGetArcs *procedure correctly expands the search graph for Algorithm 1.*

Proof Sketch. Correctness holds by observing that, by construction, Ψ at Line 27 is equal to $\mathsf{xnf}(\psi)^p|_\sigma$; putting it together with Proposition 8 and Theorem 5 of [De Giacomo *et al.*, 2022], we get the thesis. ☐

Theorem 5. *Modified Algorithm 1 with* BddBasedEqCheck *for state-equivalence checking and Algorithm 2 for search node expansion is correct and always terminates.*

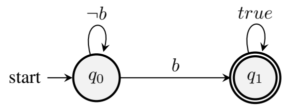


Figure 2: Minimal DFA of $\varphi = \Box a \,\mathcal{U}\, \Diamond b$

Proof. Termination follows from Lemma 1 and Theorem 4 of [De Giacomo *et al.*, 2022]. Correctness follows from Lemma 1, Lemma 2, and Theorem 5 of [De Giacomo *et al.*, 2022]. $\square$

B.2 Algorithm 1 + HashConsingEqCheck + DpllGetArcs

Theorem 4. *Modified Algorithm 1 with Equation 1 for* EquivalenceCheck *and Algorithm 2 for* GetArcs *is sound but not complete for* LTL$_f$ *synthesis.*

Proof. Soundness follows from correctness of DpllGetArcs, from the soundness of hash-consing based equivalence check, and the correctness of Algorithm 1 (Theorem 5 of [De Giacomo *et al.*, 2022]).

To disprove completeness, I show there exist a synthesis problem $(\varphi, \mathcal{X}, \mathcal{Y})$ such that the algorithm does not terminate. Let $\varphi = \Box a \,\mathcal{U}\, \Diamond b$, with $\mathcal{Y} = \{a\}$ and $\mathcal{X} = \{b\}$. The equivalent automaton of φ is shown in Figure 2. Consider any assignment with b set to false, e.g. $\sigma = \{a\}$. The repeated exploration of the agent-env move pair equivalent to σ makes the formula progression to produce ever bigger state formulas, hence making the hash-consing-based equivalence check to return false, although the associated state of the minimal DFA is always the same (q_0), see again Figure 2.

In particular, we prove by induction the following statement. Let $\varphi_0 = \varphi$ and $\varphi_n = \mathsf{fp}(\varphi_{n-1}, \sigma)$. For all $n \geq 1$, we have:

$$\begin{aligned}\mathsf{xnf}(\varphi_n) =& (((b \wedge \Diamond true) \vee \circ\Diamond b) \wedge \Diamond true)\\ &\vee (\\ &\quad \mathsf{xnf}(\varphi_{n-1})\\ &\quad \wedge (((a \vee \Box false) \wedge \bullet\Box a) \vee \Box false)\\ &\quad \wedge \Diamond true\\ &)\end{aligned}$$

Base step $n = 1$: the initial state formula in XNF form $\mathsf{xnf}(\varphi)$ is the following:

$$\begin{aligned}\mathsf{xnf}(\varphi) =& (((b \wedge \Diamond true) \vee \circ\Diamond b) \wedge \Diamond true)\\ &\vee\\ &(((a \vee \Box false) \wedge \bullet\Box a \wedge \circ(\Box a \,\mathcal{U}\, \Diamond b)))\end{aligned}$$

After applying the transformation to move to the next state, RmNext$(\mathsf{xnf}(\varphi)^p|_\sigma)$, or, equivalently, $\mathsf{fp}(\varphi, \sigma)$, we get a new LTL$_f$ formula, φ_1, that in XNF form becomes $\mathsf{xnf}(\varphi_1)$:

$$\begin{aligned}\mathsf{xnf}(\varphi_1) =& (((b \wedge \Diamond true) \vee \circ\Diamond b) \wedge \Diamond true)\\ &\vee (\\ &\quad \mathsf{xnf}(\varphi_0)\\ &\quad \wedge (((a \vee \Box false) \wedge \bullet\Box a) \vee \Box false)\\ &\quad \wedge \Diamond true\\ &)\end{aligned}$$

Note that the original formula $\mathsf{xnf}(\varphi)$ appears in the formula $\mathsf{xnf}(\varphi_0)$. Therefore, the claim holds.

Inductive step. Let the claim hold for all $i \leq n$, we need to prove that the claim holds for $n+1$. By inductive hypothesis, we have that

$$\begin{aligned}\mathsf{xnf}(\varphi_n) =& (((b \wedge \Diamond true) \vee \circ\Diamond b) \wedge \Diamond true)\\ &\vee (\\ &\quad \mathsf{xnf}(\varphi_{n-1})\\ &\quad \wedge (((a \vee \Box false) \wedge \bullet\Box a) \vee \Box false)\\ &\quad \wedge \Diamond true\\ &)\end{aligned}$$

Once applying again the same transformation but for formula φ_{n+1}, i.e. $\mathsf{fp}(\mathsf{xnf}(\varphi_n), \sigma)$, and then applying the XNF, it can be shown that we obtain the formula $\mathsf{xnf}(\varphi_{n+1})$:

$$\begin{aligned}\mathsf{xnf}(\varphi_{n+1}) =& (((b \wedge \Diamond true) \vee \circ\Diamond b) \wedge \Diamond true)\\ &\vee (\\ &\quad \mathsf{xnf}(\varphi_n)\\ &\quad \wedge (((a \vee \Box false) \wedge \bullet\Box a) \vee \Box false)\\ &\quad \wedge \Diamond true\\ &)\end{aligned}$$

note that we have a pattern: the new formula contains as a subformula the formulas computed at the previous steps.

Moreover, it can be shown that the formulas are semantically equivalent, i.e. $\varphi_n \equiv \varphi_{n-1}$ for all $n \geq 1$ (e.g. using any LTL$_f$-to-DFA tool, like Lydia), and therefore, the search will loop in the same semantically equivalent state, but on structurally different state formulas, hence without progresses of the search. See the script in `benchmark/proof-theorem-3.py` in the supplementary material. $\square$

C Empirical Evaluations

Benchmarks

I collected, in total, 1494 LTL$_f$ synthesis instances from literature, consisting of 3 benchmark families: 40 patterned instances from the *Patterns* benchmark family [Xiao *et al.*, 2021], split into the GF-pattern and U-pattern datasets; 54 instances from the *Two-player-Games* benchmark family [Tabajara and Vardi, 2019; Bansal *et al.*, 2020b], split into Single-Counter, Double-Counters and Nim datasets. Since the formulation there assumes that the environment acts first, the LTL_f instances had to be modified slightly to adapt to our setting, where the agent acts first; 1400 randomly conjuncted instances taken from [Zhu *et al.*, 2017a; De Giacomo and Favorito, 2021].

Patterns. There are 20 unrealizable GF-pattern instances, and 20 realizable U-pattern instances, constructed in the following ways, respectively.

$$GF(n) = G(p_1) \wedge F(q_2) \wedge F(q_3) \wedge \ldots \wedge F(q_n)$$
$$U(n) = p_1 U(p_2 U(\ldots p_{n-1} U p_n))$$

More specifically, G stands for $\Box$ (Always), F stands for $\Diamond$ (Eventually), and U stands for $\mathcal{U}$ (Until). The variables in the formulas are roughly equally partitioned into $\mathcal{X}$ and $\mathcal{Y}$ at random. In particular, for GF-pattern instances, the first variable p_1 is always assigned as environment variable such that all generated instances are guaranteed to be unrealizable. Moreover, for U-pattern instances, the last variable p_n $(n \geq 2)$ is always assigned as agent variable such that all generated instances are guaranteed to be realizable.

Two-player-Games. *Single-Counter* is a simple example where the behavior of the agent is completely determined by the actions of the environment. Therefore, the challenge in this family lies mostly in proving that the specification is realizable. The agent stores an n-bit counter (where n is the scaling parameter) which it must increment upon a signal by the environment. The agent wins if the counter eventually overflows to 0. To guarantee that the game is winning for the agent, the specification assumes that the environment will send the increment signal at least once every two timesteps.

Double-Counter is similar to the *Single-Counter* one, except that in this case there are two n-bit counters, one incremented by the environment and another by the agent. The goal of the agent is for its counter to eventually catch up with the environment's counter. To guarantee that this is achievable, the specification assumes that the environment cannot increment its counter twice in a row.

Nim describes a generalized version of the game of Nim [Bouton, 1901] with n heaps of m tokens each. The environment and the agent take turns removing any number of tokens from one of the heaps, and the player who removes the last token loses.

Random. This benchmark family has 1400 instances, from which there are 1000 instances from [Zhu *et al.*, 2017a], and 400 instances from [De Giacomo and Favorito, 2021]. The instances in this benchmark family are constructed from basic cases taken from LTL synthesis datasets Lily [Jobstmann and Bloem, 2006] and Load balancer [Ehlers, 2010]. Formally, a random conjunction formula $RC(L)$ has the form: $RC(L) = \bigwedge_{1 \leq i \leq L} P_i(v_1, v_2, ..., v_k)$, where L is the number of conjuncts, or the length of the formula, and P_i is a randomly selected basic case. Variables $v_1, v_2, \ldots, v_k$ are chosen randomly from a set of m candidate variables. Given L and m (the size of the candidate variable set), we generate a formula $RC(L)$ in the following way:

1. Randomly select L basic cases;

2. For each case φ, substitute every variable v with a random new variable v' chosen from m atomic propositions. If v is an environment-variable in φ, then v' is also an environment-variable in $RC(L)$. The same applies to the agent-variables.

D Plots

Here I provide the full set of experimental results.

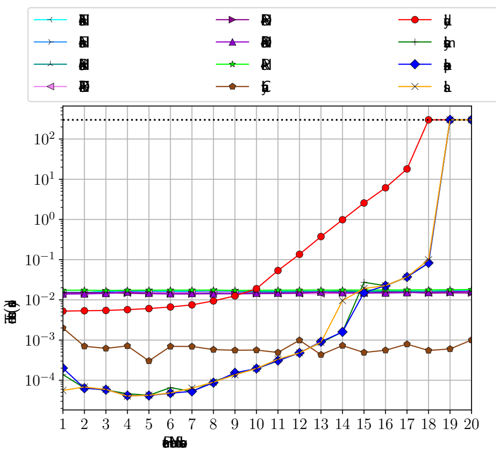


Figure 3: U-pattern.

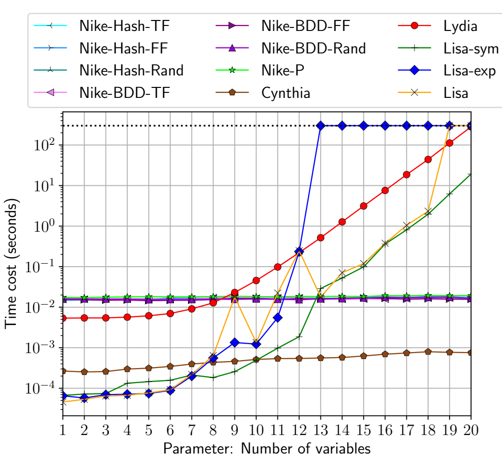


Figure 4: GF-pattern.

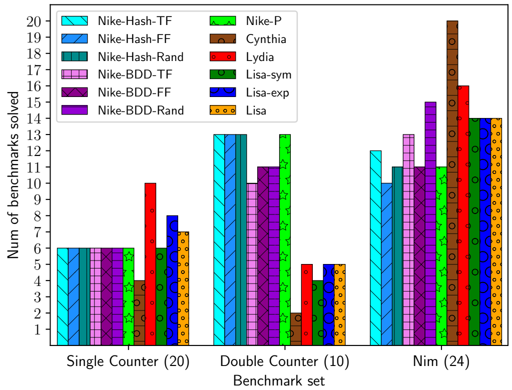


Figure 5: *Two-player-Games*.

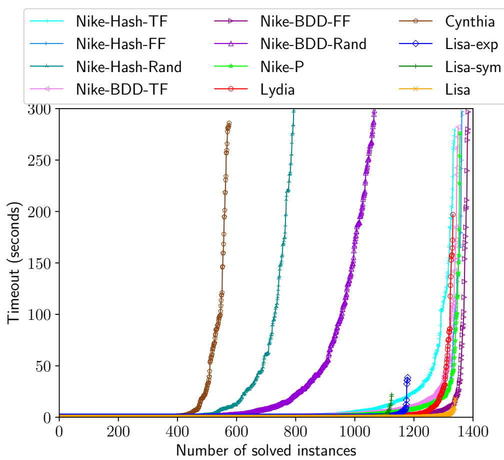


Figure 6: *Random*.

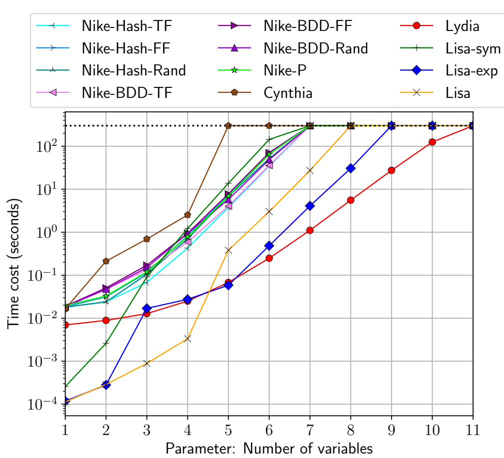


Figure 7: Single Counter.

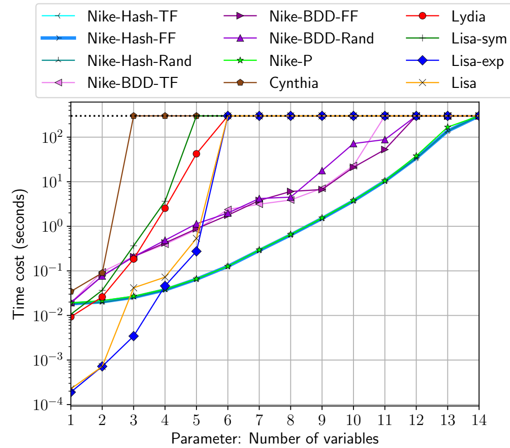


Figure 8: Double Counter.

E Results

In this section, we show all the running times for each formula of each dataset.

Table 1: Results on *Double Counter*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
counters_01	18.596958	18.268062	18.403466	19.253715	19.107806	19.412668	19.18572	34.500707	9.291831	10.6135	0.190149	0.225103
counters_02	20.013578	20.253191	19.492336	94.904176	76.403862	76.833945	21.537058	90.045937	26.0702	36.6841	0.720083	0.717793
counters_03	26.794465	25.526104	25.295872	209.794823	208.677393	207.428727	26.971165	—	184.525338	365.612	3.43736	41.6974
counters_04	38.482922	37.337264	37.016285	395.887396	423.36698	486.792135	39.26482	—	2540.111074	3641.88	45.9793	72.3087
counters_05	64.703472	65.006299	64.764067	836.840309	881.468907	1140.178069	66.213866	—	42335.932599	—	274.254	535.203
counters_06	124.922986	126.558988	127.267784	2382.959143	1774.30815	1982.499516	126.82075	—	—	—	—	—
counters_07	286.805523	286.526689	289.076264	3131.282049	3796.194518	4199.210945	296.109436	—	—	—	—	—
counters_08	635.702561	641.023385	640.855101	3890.729399	6052.720747	4515.436942	658.951354	—	—	—	—	—
counters_09	1493.792054	1499.60723	1488.196646	7136.020836	6689.333873	17742.916155	1542.665715	—	—	—	—	—
counters_10	3701.410377	3718.6171	3718.25314	23013.174684	21104.923093	71939.556943	3809.848841	—	—	—	—	—
counters_11	10280.095406	10302.711746	10233.461024	—	52653.794598	88255.826057	10521.924337	—	—	—	—	—
counters_12	33571.724906	34019.643973	33479.800756	—	—	—	38028.041671	—	—	—	—	—
counters_13	137444.743261	137462.759548	137221.411248	—	—	—	168220.51415	—	—	—	—	—
counters_14	—	—	—	—	—	—	—	—	—	—	—	—

Table 2: Results on *Gfand*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
gfand01	14.664404	15.476503	16.250197	15.496387	15.251414	16.007296	17.336509	0.266138	5.364753	0.066774	0.065305	0.045994
gfand02	15.081269	15.645543	15.272277	15.701056	15.220697	16.059728	17.27196	0.253074	5.435404	0.071976	0.057631	0.053326
gfand03	15.476061	15.80807	15.131336	16.069496	14.850944	15.798273	17.655924	0.258545	5.442908	0.074251	0.069557	0.064516
gfand04	15.985911	15.636417	16.25583	16.285712	15.024169	15.766807	17.971768	0.297777	5.679279	0.132932	0.071209	0.066989
gfand05	16.089714	15.721872	16.024327	15.85085	14.484887	15.269209	18.114179	0.313383	6.147079	0.146031	0.074355	0.076898
gfand06	15.740626	16.710326	15.815687	15.737012	14.742387	15.401362	17.966487	0.34558	6.979331	0.157342	0.088702	0.091956
gfand07	15.730463	16.658157	15.415758	15.91973	14.870688	15.4181	18.026563	0.395522	9.083914	0.212142	0.199394	0.2188
gfand08	16.389749	16.398688	15.871325	17.086864	15.363899	15.821777	18.168269	0.435402	12.807384	0.183314	0.558514	0.641949
gfand09	16.040045	16.271029	16.079457	17.072917	16.543945	15.585794	18.624364	0.460944	23.038721	0.257216	1.34533	17.8815
gfand10	15.999361	15.719367	15.88747	16.66938	16.642694	15.92157	18.241217	0.51344	45.503634	0.48844	1.22901	1.33807
gfand11	16.556774	15.795863	16.99089	16.948603	15.476833	15.640557	18.007788	0.540068	98.035326	0.976966	5.52441	22.4268
gfand12	16.112348	15.954323	16.938068	17.02628	16.220457	15.255452	18.40826	0.542533	221.638825	1.87119	235.082	247.046
gfand13	16.312971	16.503906	16.578508	15.655115	15.743175	16.046211	18.336155	0.558839	516.309332	28.8656	—	17.8299
gfand14	16.554934	16.482217	16.076889	15.698039	16.815838	16.176841	18.573829	0.574802	1273.111451	52.5136	—	70.9903
gfand15	17.415826	16.363838	16.67325	15.840495	16.912334	16.859937	18.297912	0.625352	3132.349671	98.9009	—	119.025
gfand16	17.963636	16.808938	17.181037	15.809993	16.506084	17.695604	19.236552	0.692731	7597.368526	361.028	—	376.954
gfand17	17.364304	17.217969	16.823467	16.197285	15.790041	17.33682	19.108987	0.740725	18622.289185	811.707	—	1046.99
gfand18	17.597947	18.341532	17.178884	16.768091	16.089661	17.332376	19.336902	0.796435	44042.081704	1966.76	—	2329.31
gfand19	17.501234	18.12741	16.667382	16.797178	15.800645	17.688795	19.150142	0.770764	112126.560954	6231.58	—	—
gfand20	17.96385	16.980853	16.880816	17.293062	15.652091	16.575834	19.382017	0.754719	280759.10069	19257.5	—	—

Table 3: Results on *Nim 01*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
nim_01_01	21.652504	21.852993	22.425201	22.51097	22.575411	22.619657	22.869925	10.701386	9.243031	0.511151	0.054589	0.039505
nim_01_02	36.510922	32.635422	29.130419	51.644759	46.93518	42.953328	33.937856	34.209743	14.044925	0.783896	0.107531	0.108793
nim_01_03	124.350392	748.812837	71.802001	147.81038	500.318885	94.538401	124.496853	50.863318	28.067614	1.39775	0.165809	0.159741
nim_01_04	655.990604	5078.174691	346.277822	478.273479	3572.571717	175.8517	636.957444	51.149021	148.985605	1.90265	0.253887	0.216297
nim_01_05	3951.684383	31716.609289	2088.068016	1893.314961	23764.470102	322.293758	3939.678555	83.136903	648.186875	2.67965	0.354514	0.33663
nim_01_06	25804.192511	185198.807923	14973.912576	8558.792909	146791.191633	607.119617	27804.395652	117.085951	2835.283053	4.26119	0.491802	0.509569
nim_01_07	180527.335506	—	104195.615515	40304.12107	—	1784.890989	106272.648966	173.173464	12656.386116	6.0989	0.713635	0.713666
nim_01_08	—	—	—	191314.266217	—	5567.172292	—	223.023923	49118.665584	—	—	—
nim_01_09	—	—	—	—	—	21620.10071	—	382.891399	137688.406851	—	—	—
nim_01_10	—	—	—	—	—	66215.040678	—	327.374822	—	—	—	—
nim_01_11	—	—	—	—	—	—	—	622.210931	—	—	—	—
nim_01_12	—	—	—	—	—	—	—	532.105444	—	—	—	—
nim_01_13	—	—	—	—	—	—	—	641.369547	—	—	—	—
nim_01_14	—	—	—	—	—	—	—	905.842151	—	—	—	—
nim_01_15	—	—	—	—	—	—	—	1110.560059	—	—	—	—
nim_01_16	—	—	—	—	—	—	—	1243.463515	—	—	—	—
nim_01_17	—	—	—	—	—	—	—	1610.98619	—	—	—	—
nim_01_18	—	—	—	—	—	—	—	1699.635879	—	—	—	—
nim_01_19	—	—	—	—	—	—	—	2064.899084	—	—	—	—
nim_01_20	—	—	—	—	—	—	—	2631.465242	—	—	—	—

Table 4: Results on *Nim 02*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
nim_02_01	86.975632	112.611648	105.373406	98.569137	119.114981	119.995565	114.925515	115.180326	22.998994	7.71421	0.335915	0.32857
nim_02_02	14861.324807	16186.670998	14325.332997	7333.083561	8301.186367	7004.188455	16339.154739	730.429379	291.899614	42.5965	1.10907	1.10246
nim_02_03	258217.922498	—	—	107711.86067	200600.7252	168324.776357	—	1018.461399	4443.044252	178.374	2.8325	2.90445
nim_02_04	—	—	—	—	—	—	—	5464.305582	73887.195858	287.6	6.57803	7.06112
nim_02_05	—	—	—	—	—	—	—	22790.095015	—	—	—	—
nim_02_06	—	—	—	—	—	—	—	37507.865014	—	—	—	—
nim_02_07	—	—	—	—	—	—	—	23257.818323	—	—	—	—
nim_02_08	—	—	—	—	—	—	—	169177.832518	—	—	—	—
nim_02_09	—	—	—	—	—	—	—	95835.707556	—	—	—	—
nim_02_10	—	—	—	—	—	—	—	—	—	—	—	—

Table 5: Results on *Nim 03*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
nim_03_01	14589.428787	15583.749344	17541.842061	7563.240427	7376.988988	8799.891142	15761.988348	650.571523	270.715502	473.228	0.26237	0.27006
nim_03_02	—	—	—	—	—	—	—	12655.221579	42245.551844	3102.4	24.4585	8.01435
nim_03_03	—	—	—	—	—	—	—	—	—	—	—	—

Table 6: Results on *Nim 04*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
nim_04_01	200248.683865	229673.2003	251716.519243	80051.226611	99057.523944	83007.538629	261215.876355	8614.624562	19574.724446	6523.22	6.79701	22.7605
nim_04_02	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_03	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_04	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_05	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_06	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_07	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_08	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_09	—	—	—	—	—	—	—	—	—	—	—	—
nim_04_10	—	—	—	—	—	—	—	—	—	—	—	—

Table 7: Results on *Nim 05*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
nim_05_01	—	—	—	—	—	—	—	220651.046985	—	—	—	—
nim_05_02	—	—	—	—	—	—	—	—	—	—	—	—
nim_05_03	—	—	—	—	—	—	—	—	—	—	—	—
nim_05_04	—	—	—	—	—	—	—	—	—	—	—	—
nim_05_05	—	—	—	—	—	—	—	—	—	—	—	—
nim_05_06	—	—	—	—	—	—	—	—	—	—	—	—
nim_05_07	—	—	—	—	—	—	—	—	—	—	—	—
nim_05_08	—	—	—	—	—	—	—	—	—	—	—	—
nim_05_09	—	—	—	—	—	—	—	—	—	—	—	—

Table 8: Results on *Random Lydia Case 03 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	16.713829	17.150155	20.492264	17.892924	18.443531	19.430784	25.886465	14.689382	12.776132	0.06382	0.144459	0.151035
2	264.680185	114.455337	3557.0157	313.911392	84.343603	1209.118128	120.310362	257052.503168	359.320861	1.93298	1014.14	4.331
3	234.630113	128.765479	22201.139848	298.172512	96.743202	6647.968904	135.570123	—	2123.071477	3.285	6682.4	9.94998
4	19.079327	19.830259	25.699179	18.866273	19.797592	25.574912	22.010723	13.133271	7.249208	0.040258	0.030786	0.031127
5	52.336247	19.384262	100.87368	49.863463	20.542235	84.771026	22.809315	689.867092	12.085266	0.140104	0.314111	0.275143
6	1577.386535	2093.895391	—	725.614422	200.807375	1511.57062	2132.3953	—	8.397469	0.057513	0.02679	0.022097
7	16.018741	16.307278	16.355825	15.962757	15.742274	16.11696	17.791634	0.334175	6.456763	0.061564	0.037286	0.034532
8	18.729788	17.292283	21.160177	17.42736	16.338228	21.614487	19.61433	18.958183	8.408105	0.042382	0.041671	0.037531
9	643.679594	33.954107	446.159708	440.506309	24.064403	331.912445	37.463396	1446.110168	9.473982	0.108847	0.299497	0.223697
10	271.966483	125.269605	6986.377843	291.202488	88.268931	1212.15484	132.242892	—	405.910661	1.5725	1043.94	4.32516
11	16.356166	16.172018	16.024578	16.135849	15.641622	15.973239	18.363988	0.350353	6.698781	0.062039	0.043551	0.042436
12	17.644318	17.387841	21.174377	18.001583	16.112062	22.807826	19.799071	14.507595	7.888645	0.049974	0.038047	0.035527
13	458.727969	650.151788	29907.015123	245.595608	124.387906	832.017953	664.569816	194854.172797	7.462804	0.059772	0.022898	0.020817
14	16.357051	15.783832	16.127345	16.127328	16.171758	16.165932	17.972411	0.361341	6.051805	0.063003	0.035106	0.033882
15	15.077766	14.872916	16.035513	14.784308	15.072014	15.929674	18.103903	0.337413	6.078458	0.060975	0.03632	0.036488
16	14.832038	14.856293	14.985044	15.224145	15.43672	16.071333	17.977368	0.315692	6.061586	0.062107	0.039855	0.036139
17	190.618258	121.755879	9383.936116	250.605727	92.910674	2885.723867	129.325476	159589.187539	614.59472	26.9009	5348.79	19.5155
18	55.334547	18.848157	165.1421	45.913533	19.035935	38.694017	21.108721	104.673617	9.352371	0.044335	0.07384	0.058788
19	54.427956	18.188667	153.036554	45.203281	18.522585	37.837535	20.588738	140.255972	9.413107	0.058049	0.065438	0.057449
20	16.356937	15.976384	17.81288	15.462054	16.709166	17.513119	19.132442	0.669631	50.897121	0.269313	6.2213	5.67827
21	17.074823	15.546332	17.026238	15.215205	17.120348	16.684478	19.173544	0.689072	43.202003	0.248035	5.72242	5.48191
22	724.066042	225.894922	9607.277385	390.582355	162.926601	4208.671728	231.761959	18162.542336	208.620993	1.36026	7.8407	1.16876
23	16.670814	16.568445	16.375378	15.778166	15.935098	16.151078	17.966399	0.342892	6.250231	0.060198	0.043821	0.036132
24	1130.928125	35.791545	865.166314	701.890274	25.626445	571.691509	37.252717	2191.079127	14.486889	0.139222	1.00649	0.870784
25	1444.467959	1923.396813	45439.840468	474.863318	189.548111	787.300671	1949.683635	—	8.203597	0.048801	0.022684	0.021775
26	2404.261809	53.057997	1608.240944	1430.363825	32.974939	950.69129	55.344948	1807.668227	18.198376	0.19075	1.49014	1.33095
27	16.597258	15.653909	16.14442	16.231644	15.828257	16.152835	17.970229	0.334474	6.294527	0.059986	0.03996	0.035734
28	55.680205	17.353568	158.505392	44.872868	18.154151	34.895507	20.847353	105.016481	10.106468	0.042887	0.065826	0.059607
29	17.522887	15.841538	17.748955	15.741463	16.108436	16.981958	19.196889	0.769509	15.969278	0.237764	0.642442	0.573971
30	17.117795	17.075452	18.065027	16.628681	17.304774	17.937307	20.331448	1.155465	9.208742	0.030732	0.02721	0.024142
31	1902.780586	62.317975	1220.678367	1043.548058	49.56786	694.689226	66.300423	1692.896715	15.93553	0.282741	1.97019	1.83755
32	1576.045705	2026.958503	—	724.499705	153.352981	1507.332282	2067.653725	—	7.977131	0.055556	0.02419	0.023034
33	17.305131	18.800291	21.397722	17.852816	18.078376	21.178684	20.37338	14.411195	10.028272	0.052239	0.060712	0.049502
34	19.28371	17.723689	21.082703	18.26937	16.543205	20.983076	19.283711	27.988163	8.191364	0.046862	0.04448	0.038248
35	17.493864	18.623928	17.144722	16.675358	16.537226	17.749155	20.034379	1.088371	9.066229	0.029958	0.030354	0.03343
36	51.766919	20.608086	99.927493	48.562024	20.013443	84.912427	22.961859	683.649909	11.487724	0.140631	0.346283	0.267817
37	245.469889	135.929817	36008.074847	302.795444	99.122248	6783.819619	142.800676	—	2041.623783	2.45021	5084.13	5.07294
38	627.606506	114.970183	5609.496118	546.581704	85.3683	1212.832145	120.850876	85324.327072	542.216844	1.83944	1028.28	2.72172
39	16.477352	16.537701	16.103698	15.758982	15.666638	15.347946	18.063974	0.358671	6.32582	0.062576	0.038331	0.040518
40	218.775557	128.720692	16199.217323	268.852499	93.874143	3984.103628	134.270809	218541.788047	844.145195	1.77382	1820.05	5.68449
41	168.578158	127.149589	4213.349162	200.99304	93.129704	1880.764779	133.278853	72265.733766	455.314357	0.67818	317.863	1.65207
42	16.34758	16.364573	16.095463	15.934677	15.327764	16.075688	18.314594	0.366722	6.362349	0.060675	0.039911	0.035567
43	16.009279	15.007877	15.280716	14.554233	14.67194	16.143937	18.59881	0.300164	6.587226	0.06028	0.037458	0.036005
44	16.666632	18.749467	19.583271	16.371975	17.640385	21.214213	20.303502	14.702622	9.606057	0.050511	0.057788	0.052592
45	17.809862	17.518336	19.831975	17.816912	16.62885	19.481835	19.55468	26.557034	8.779915	0.042701	0.046153	0.037496
46	16.471657	16.187371	16.351982	15.508155	16.045107	16.353617	18.925974	0.591926	42.797751	0.301773	7.71272	7.37132
47	1082.606504	63.88079	806.591752	644.893042	47.786146	514.973699	65.520798	1417.136701	16.362788	0.246407	1.61334	1.57232
48	17.886467	17.558697	16.986374	16.678409	16.920717	17.003458	19.298747	0.647124	45.319374	0.214015	5.75524	5.58755
49	16.540877	15.247747	15.57153	16.127498	15.87803	16.005635	17.941164	0.278557	6.589328	0.061237	0.038313	0.034757
50	15.734072	16.636959	16.035176	16.039607	16.350442	15.479667	18.612241	0.428451	6.234186	0.026722	0.02272	0.021039

Table 9: Results on *Random Lydia Case 04 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	16.870001	17.153821	17.18682	16.384968	17.227759	16.543762	19.17354	6886.058277	11.739812	0.328307	0.520087	0.516529
2	15.399163	15.196455	15.486768	16.118445	15.354612	15.305571	18.010372	0.384879	6.501435	0.058177	0.034233	0.032827
3	247.406245	18.096118	1644.213418	178.924989	18.780541	138.306082	20.774158	2303.136716	23.831855	0.089467	0.356991	0.337432
4	1773.638344	691.986975	—	1593.325115	495.672686	—	711.480355	—	156043.561906	83.2831	—	1869.97
5	3697.624058	578.107085	89662.888683	1848.740326	462.946393	31916.684263	587.466884	—	1902.637411	6.17162	495.238	10.5653
6	19.20774	18.906754	19.334721	18.794765	19.034083	18.790621	20.62917	1.393927	10.059809	0.034501	0.027884	0.026008
7	1033.160623	705.34598	—	1102.201835	500.411941	—	720.373232	—	196991.230935	79.4626	—	1836.46
8	16.276201	16.557357	15.570412	16.050833	15.857689	16.445607	18.30212	0.427595	6.801465	0.062123	0.041645	0.043245
9	16.220484	15.344821	16.726388	15.388823	14.90343	15.133476	17.488743	0.307166	6.335313	0.065033	0.037991	0.036245
10	528.952816	542.025474	115056.660334	661.978156	403.736946	9262.388932	553.574231	—	12009.676309	24.8063	—	129.708
11	4113.077586	4912.778258	—	1255.594826	501.361131	8738.39105	4948.390074	—	8.427913	0.101292	0.070586	0.023958
12	18.060112	19.582205	19.232682	18.712173	19.218014	20.761301	20.54302	80.191847	22.788498	0.083626	0.230281	0.209495
13	63.204606	19.646581	184.887076	51.204644	17.818737	51.350984	20.992928	66.253604	9.812948	0.061691	0.072654	0.059898
14	18.59421	19.17194	18.983517	18.617359	17.241058	17.911433	20.089896	1.615347	10.072841	0.035515	0.033486	0.025501
15	16.431986	17.879536	17.782248	15.871638	16.036575	16.999188	18.919838	1.322599	57.378305	1.33781	44.52	41.7204
16	1162.705726	1194.294394	—	1329.430143	691.836271	94842.901311	1217.434627	—	21501.88651	25.8822	—	410.237
17	269.829022	155.577164	18432.587722	312.423837	123.297704	10050.661301	159.901717	—	1417.155195	0.76009	616.448	1.55039
18	20.951444	20.832115	21.102809	21.138189	21.160877	21.594988	22.86222	1778.805749	10.187241	0.244391	0.345002	0.311715
19	26718.268166	33774.276741	—	6277.411501	1100.814974	14089.571064	—	—	8.971867	0.091563	0.024193	0.021914
20	6645.128291	610.843628	—	5575.774526	439.373057	—	619.993118	—	128234.709265	77.2975	—	1843.85
21	813.005043	585.334113	175082.184445	780.785159	419.634689	10393.307308	592.314906	—	8923.181014	44.582	—	213.586
22	41499.589367	438.825398	26221.993615	15704.163416	367.840986	10694.960992	443.137796	—	61.736103	1.37953	46.7087	47.8082
23	20265.716712	2155.098302	—	8052.62209	1329.331051	149141.397963	2171.736018	—	2381.60708	7.0781	441.055	11.5326
24	1443.519399	635.212743	—	1398.462387	456.11369	79449.528756	642.633105	—	25241.345964	72.8237	—	301.372
25	17.510523	17.345556	17.68764	17.017076	17.424828	17.2006	18.479083	0.642831	6.814329	0.031561	0.025388	0.023043
26	26442.212578	33376.556292	—	9237.860156	859.084715	26441.503646	33894.608053	—	8.69352	0.149138	0.025823	0.024636
27	23.420571	17.763076	34.999837	23.525474	18.395255	36.92474	19.386776	285.641275	11.818623	0.103162	0.122483	0.10628
28	245.715633	19.360676	1632.002209	180.74237	19.675706	138.855659	21.026316	1803.642676	20.473892	0.085041	0.370282	0.282473
29	12853.634445	2369.522389	—	5158.494929	1426.899595	186948.62404	2375.277667	—	5814.895464	17.8198	1668.06	28.5502
30	17.394228	17.155361	17.567925	17.377432	16.444314	17.969385	18.47071	0.677391	6.538339	0.029614	0.028862	0.023412
31	16.428081	16.555444	16.75394	14.817397	14.928218	16.095582	18.073823	0.35766	5.94707	0.069534	0.040131	0.033822
32	1389.531201	2482.372896	—	1478.682314	1345.482246	—	2526.51656	—	157837.988954	85.7016	—	2768.92
33	18.930979	19.433514	20.556808	19.492616	20.067001	20.728699	20.768037	78.942834	26.084775	0.080169	0.273666	0.219388
34	17.806737	23.800582	29.118223	19.591375	24.045317	28.318308	25.41602	72.058796	13.683566	0.063089	0.104555	0.089007
35	43632.699118	288.109294	29699.15258	19383.436399	199.104787	14087.576926	291.612039	—	103.261976	1.48659	55.8196	53.549
36	16.27677	16.394562	16.304074	16.396582	16.538878	15.832221	18.144872	0.416398	6.518815	0.06577	0.040485	0.0322
37	16.924209	15.82008	17.503772	16.962768	17.113509	16.672981	18.824079	15378.321264	10.891294	0.321595	0.706819	0.510498
38	17.115121	16.137143	16.539814	15.700243	16.514551	15.579854	18.972824	1.206223	19.45839	0.516172	1.33028	1.24466
39	854.180186	581.279795	94855.865723	942.913678	440.54094	30655.329388	597.542584	—	7129.672146	3.40026	—	23.4741
40	1019.26628	1000.59163	—	1181.002998	730.500061	—	1025.689427	—	136656.758395	86.7387	—	1901.53
41	7411.949775	9596.189369	—	2851.782378	745.681609	14062.726104	9696.939732	—	8.965806	0.161929	0.068553	0.029518
42	62313.335697	219.527225	39866.466445	29989.033676	102.97094	19846.589668	222.06792	—	132.329979	1.12276	40.5966	32.9678
43	1102.776587	700.368441	—	1149.92034	494.607613	—	725.440218	—	49492.006869	25.9512	—	175.126
44	2735.065293	597.388931	—	2653.892219	441.051312	72447.476077	617.076613	—	22126.671165	412.83	—	420.189
45	16.337958	16.572393	15.799223	15.690027	16.085439	16.515104	18.014188	0.387981	6.91398	0.056931	0.087155	0.03429
46	245.819861	18.612222	1674.154767	177.601241	17.744922	153.350328	21.046032	2256.588951	20.86646	0.08617	0.394199	0.289619
47	2606.001119	563.022806	119788.015925	2555.982148	437.743402	18093.501345	582.170434	—	4168.452764	11.7834	—	102.313
48	18725.787665	111.899406	17564.597023	9594.940835	57.954708	9157.838574	114.906748	—	92.187092	0.614829	13.2141	13.0344
49	18.731715	19.799915	19.142488	19.450474	19.377172	19.130106	21.210979	1.688165	9.751101	0.041184	0.029841	0.027085
50	23.651542	18.321266	36.24551	24.950433	19.040056	38.544705	20.051343	84.997919	11.695353	0.069393	0.072315	0.069581

Table 10: Results on *Random Lydia Case 05 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	17186.954282	2956.285242	—	5894.234357	1710.078046	—	2955.30058	—	21309.590105	78.1458	—	61.445
2	5590.052495	5562.885684	—	5285.853787	3710.628698	—	5620.573055	—	164366.265304	40.5712	—	2756.39
3	19.681307	19.744672	19.845728	19.39953	18.779845	19.490557	21.201815	1.897343	11.232309	0.041901	0.072184	0.028203
4	1302.535106	20.075408	—	909.670963	19.545558	656.361316	21.515379	4480.153605	93.623848	0.263688	2.7205	1.55379
5	5573.333707	3757.327975	—	5286.536226	2344.961407	—	3817.344839	—	—	1439.49	—	3940.64
6	—	22450.212368	—	107409.098376	10054.463637	—	25579.55604	—	65789.442762	147.514	—	209.235
7	16.814876	16.623774	16.986842	16.689582	16.455677	17.189713	18.338521	2.025441	9.801064	0.068553	0.077106	0.036585
8	—	2126.93578	—	—	1849.490522	278185.208134	2130.760887	—	600.227128	15.8775	6478.4	5905.99
9	20.512661	22.927794	64.443445	21.950302	22.727967	67.347209	23.988407	61.845556	23.771483	0.14032	0.254064	0.206482
10	41379.737563	4296.541372	—	30986.878063	3062.460679	—	4309.954435	—	152900.295309	93.7073	—	836.937
11	—	—	—	101347.380861	6606.700134	—	—	—	12.704127	0.395275	0.072536	0.027598
12	1418.814216	3159.765087	—	1643.036592	1954.006579	—	2601.183561	—	—	280.439	—	2443.92
13	16.115339	16.85105	16.872398	16.594	15.526733	16.537043	18.220026	0.910411	7.229879	0.058309	0.038585	0.034353
14	15.839468	16.225425	16.077097	15.914361	15.451518	15.869562	18.891069	1.137805	7.015719	0.038749	0.022997	0.023098
15	18.412611	22.08611	34.559584	19.534496	21.375944	35.790522	24.273989	53.907668	48.817792	0.139971	0.573041	0.451648
16	16.884644	16.511943	16.838154	16.205437	15.393086	17.48339	18.951204	6143.951754	26.12753	0.709863	8.6393	8.00446
17	29.416442	16.756095	52.080831	25.779051	16.289428	49.652242	19.294186	145.731407	9.268959	0.09791	0.04508	0.043644
18	6221.6962	4671.930719	—	5607.917006	3359.309386	—	4725.680703	—	—	98.7839	—	708.63
19	1774.845944	22.950869	11109.381353	254.890304	33.410932	648.356577	24.981649	31946.966309	23.33411	0.51877	1.78989	1.52522
20	18.882366	18.591237	20.143894	19.832969	19.205332	21.577407	20.883271	25.432817	25.344135	0.140533	0.25916	0.191945
21	16.390954	16.533302	15.093798	15.280234	16.225935	15.436742	17.547037	0.378743	6.847737	0.063052	0.035091	0.031177
22	13222.516209	3222.785388	—	12745.825357	2128.946567	—	3268.990053	—	—	665.876	—	828.312
23	5791.618965	3529.702718	—	5338.78545	2196.487077	—	3579.716612	—	—	1234.47	—	—
24	60126.795195	4629.063863	—	21839.61151	2895.27473	—	4653.241325	71532.593849	3761.347561	18.7532	381.019	15.3727
25	17.926225	17.638031	18.533645	17.941805	17.705119	18.56386	19.374265	0.930027	2777.283162	6.39642	—	46.9125
26	5970.406071	3698.01491	—	5527.909497	2235.609818	—	3719.965382	—	—	611.597	—	9439.84
27	16.551957	16.117154	16.883687	15.987323	15.827763	17.022383	17.562635	0.518561	10.19297	0.073611	0.083815	0.034935
28	1297.068977	18.534239	—	910.295872	18.934604	598.199571	21.436888	4651.0617	95.375922	0.267702	1.73283	1.49475
29	16.660695	15.739375	18.001215	17.289918	15.823626	17.447642	18.287535	0.709812	7.816516	0.038588	0.02606	0.023184
30	15.918118	14.8817	16.071011	16.489833	15.062531	16.148014	17.626189	2.013265	8.585718	0.058819	0.036582	0.035317
31	30.753644	16.92691	69.102268	27.195629	16.637807	69.568523	19.441971	260.487562	13.271986	0.296385	0.457966	0.383364
32	1301.09532	18.499171	—	906.720225	18.259747	660.136974	21.145515	4317.825619	91.79369	0.270065	1.81145	1.47449
33	19.190267	17.510867	28.405232	18.846629	18.294869	28.318513	20.730417	70.614554	105.392919	0.222285	1.36502	1.15861
34	13555.588058	8004.234145	—	11507.609134	3970.073312	—	8126.622796	—	—	4665.3	—	14010.2
35	19.126195	19.446532	25.934593	19.62422	20.307674	28.873681	20.739317	67.700472	105.083548	0.2681	1.35202	1.21066
36	16.302327	16.458631	16.697228	16.660161	15.271723	16.284911	17.760457	0.435088	8.443836	0.070994	0.041034	0.032624
37	21031.281618	8841.655426	—	20069.762917	6513.826754	—	8935.08687	—	—	102.137	—	2100.7
38	2194.047751	37.467729	63045.095541	1423.494421	47.179014	11487.086893	39.11185	—	76.663057	1.25485	28.1663	27.782
39	2484.435059	1064.25146	—	2699.596699	800.446284	—	1075.987186	—	—	162.197	—	704.192
40	17.527207	18.074305	18.003238	16.743924	16.657964	18.139553	18.797736	3248.030091	22.562076	0.706082	3.23709	2.90122
41	16.502233	16.718477	16.874013	14.90001	15.296187	16.402172	17.803392	0.767401	8.225756	0.071777	0.037154	0.030856
42	—	25559.305895	—	114584.540829	10563.332347	—	36410.089268	—	—	309.986	—	326.875
43	28.18697	18.240519	77.41634	26.48372	18.509657	58.826588	19.411762	238.788281	13.613655	0.326448	0.55718	0.439837
44	15.129076	16.485855	16.333648	15.323366	14.960551	16.085247	18.041135	0.426725	7.892468	0.052285	0.036521	0.03136
45	25.03671	23.993182	24.121116	25.558701	25.497336	26.373268	27.030378	25726.927969	21.329737	0.629934	2.87334	2.61447
46	—	615.799344	—	—	270.196747	—	619.948209	—	839.212319	12.6882	5595.98	5229.77
47	6192.634582	3489.31044	—	6409.673235	2196.870227	—	3530.202773	—	—	654.063	—	6858.53
48	18.235069	17.583504	18.60941	18.584168	18.284128	18.179705	19.468825	2.442147	168.732547	10.075	1012.31	862.443
49	—	2492.752344	—	—	2097.715188	287512.104004	2428.965454	—	752.767819	21.4778	8183.0	7540.64
50	19.71022	19.845304	20.27337	19.657116	19.781079	19.682149	20.862223	2.264566	12.563414	0.048344	0.033352	0.029265

Table 11: Results on *Random Lydia Case 06 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	17.725612	17.815749	18.306462	17.260368	16.949622	18.316431	26.258953	1.824316	74788.499671	128.161	—	1605.91
2	17.622506	18.725651	28.870486	20.297471	19.637225	29.361514	21.549324	88.152674	108.293412	0.449279	1.11059	0.61805
3	2647.044011	39.104455	147931.751387	1283.700043	50.225297	15786.606906	42.017331	—	112.983522	1.57557	48.2856	46.751
4	16.78076	16.583245	16.804801	16.658298	15.80621	16.467267	18.624064	2.675993	10.65112	0.043024	0.045095	0.047252
5	—	—	—	—	—	—	—	—	75795.185641	198.599	—	312.79
6	12320.366361	64.441768	—	5132.779359	42.093978	3864.07905	67.630661	91133.936828	25.545632	0.353424	0.384728	0.318895
7	—	—	—	—	32863.42443	—	—	—	18.455482	1.44071	0.030167	0.024532
8	17.82573	18.175483	17.733877	17.910561	17.758118	18.076	20.096125	12947.866661	41.300468	1.15366	14.6333	13.8457
9	20.053167	20.435898	84.134085	20.002887	19.924446	87.878398	22.500527	432.876722	582.385469	0.896769	12.7224	0.897712
10	17.716032	17.280368	17.479329	17.406261	17.582699	18.81597	21.096807	3.851121	2985.885631	555.273	—	277.605
11	—	—	—	141181.279957	8402.491324	—	—	—	13.053026	1.64984	0.074758	0.027474
12	17.024589	16.998167	16.840117	16.611187	16.979116	17.085917	19.135059	0.48344	10.563795	0.044445	0.061931	0.046672
13	17.040205	17.126015	18.244227	19.067354	16.74355	18.86702	20.539393	1.033106	56172.697453	72.7671	—	892.311
14	26301.516771	4032.275555	—	23596.600669	2386.960475	—	4174.430387	—	—	3158.18	—	—
15	19834.802467	18286.26178	—	19721.813078	10996.921383	—	19076.639896	—	—	802.983	—	—
16	24734.504829	18775.691775	—	23753.583476	11854.909836	—	19221.525103	—	—	6818.89	—	—
17	16.6766	16.073141	16.66132	16.910015	15.913763	17.014416	18.730441	1.855938	11.207786	0.0757	0.098661	0.076674
18	—	63137.169839	—	—	24601.669398	—	64107.643242	—	—	476.795	—	692.795
19	113698.055349	17913.488071	—	110763.847018	10665.933706	—	18487.432555	—	—	9240.79	—	—
20	19.316355	18.677881	18.85099	18.713745	18.340243	18.273598	20.884063	3.772515	2671.076192	438.715	—	485.7
21	7140.714042	20.763188	—	4890.0597	20.828187	3178.002565	23.034582	119017.441801	481.347471	0.948815	28.8825	1.0312
22	19.716807	19.342622	85.046277	20.624832	19.764781	86.692778	21.914121	413.545907	573.00698	0.799826	15.236	0.904199
23	60.401429	20.955882	289.653173	65.214642	22.400495	329.55547	24.78023	1663.157691	27.59101	0.227142	0.29925	0.234439
24	17448.336731	17950.574591	—	18453.006294	10811.445129	—	18804.099254	—	—	2960.7	—	1010.1
25	15800.07337	61.71774	—	8147.754581	74.657259	65916.391457	65.441964	—	379.027379	5.55059	622.332	12.179
26	20.08871	20.679686	20.142815	20.916778	20.563588	21.078182	22.385438	2.627018	13.940041	0.088197	0.03387	0.027611
27	—	179686.597171	—	—	80035.726928	—	—	—	—	—	—	1431.36
28	2866.200072	62.329039	283871.96806	1809.079596	72.992599	24161.619692	64.889624	—	287.736236	3.14671	263.849	8.40822
29	19.171152	19.505468	47.074791	20.48262	20.581259	48.704523	21.824145	37.203601	26.545113	0.271858	0.265133	0.182623
30	5840.081411	37.802951	—	3737.900743	29.260213	3220.039076	41.598124	86151.022195	464.287332	0.523643	9.86331	0.558945
31	17.692214	17.293052	17.869315	18.837028	18.073613	18.026609	20.912307	3.117526	1865.090784	190.66	—	288.638
32	15687.920867	60.255655	—	8155.123249	75.025749	87413.367931	64.013807	—	363.573356	6.09257	584.65	11.9831
33	14479.991594	15942.799789	—	14992.214685	9756.641969	—	16598.157156	—	—	640.662	—	2155.99
34	1503.76944	21.15171	—	1017.000229	20.726429	786.887448	22.632066	7619.817101	90.255739	0.478207	2.09661	1.91511
35	49.464049	18.113441	163.680663	52.076523	18.765431	195.186827	20.344538	12373.527089	22.395965	0.762474	1.21756	1.07333
36	185338.515399	25493.089778	—	172516.614662	16917.052234	—	26413.000282	—	—	22070.0	—	—
37	116711.256208	17780.586724	—	111141.43535	10696.312535	—	18668.029405	—	—	3227.14	—	—
38	7455.26956	115.523274	—	4742.474649	61.038613	4635.772466	125.081206	66214.901623	74.589504	0.310648	0.889523	0.825519
39	64716.590774	81279.694386	—	50584.270968	36672.849229	—	86197.115618	—	—	13047.1	—	—
40	18.643652	18.7264	18.533985	18.687562	18.960498	18.534082	21.188701	3.204172	1152.111611	94.2117	—	71.3779
41	152140.416259	118311.629543	—	87395.480084	42197.762605	—	—	—	—	20398.7	—	—
42	—	—	—	185133.314594	9858.934197	—	—	—	13.112195	0.54108	0.070434	0.074639
43	2209.962737	37.645791	36344.135408	1385.541474	48.208086	9607.154131	40.748249	—	78.137948	1.16236	28.3838	27.7359
44	—	2085.218596	—	—	1399.316105	—	2111.386651	—	4696.974627	33.363	—	98.6493
45	18.116922	17.628702	17.743739	18.203033	17.896056	17.510276	19.911393	0.977629	7.911422	0.043721	0.027849	0.024403
46	69705.083347	20110.527227	—	59295.14942	12239.06974	—	20880.078621	—	—	1271.97	—	—
47	19388.081826	26454.657244	—	19933.131137	15256.891156	—	20528.079671	—	—	666.857	—	—
48	21.17218	20.680712	20.582413	19.765074	21.099256	20.606945	22.691593	2.555701	15.833586	0.055275	0.081228	0.029366
49	17.024565	18.186843	16.208681	16.057734	17.619484	16.304216	19.621447	0.963113	8.187546	0.049854	0.025115	0.023008
50	18.11862	18.765472	18.534884	16.789009	17.716776	18.277607	20.5402	3.287861	4724.58507	345.182	—	181.636

Table 12: Results on *Random Lydia Case 07 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	—	139659.636414	—	272613.224302	86346.68118	—	150617.396927	—	—	6046.44	—	—
2	108246.403146	115.779725	—	46391.418416	191.093623	—	118.8012	—	1876.379831	27.1607	12457.7	80.1553
3	—	—	—	—	—	—	—	—	—	—	—	—
4	17.566906	17.394854	17.141297	17.420313	17.46057	16.627154	18.922813	0.912246	17.759666	0.055737	0.143472	0.120822
5	136.994026	23.104773	914.861403	172.446904	24.219817	1016.665293	25.182862	21372.884565	130.019039	2.96725	7.03019	6.27644
6	17.232201	16.426434	17.360178	16.669022	15.882919	16.582951	18.614869	0.845496	10.277482	0.034572	0.057241	0.04242
7	32592.043851	37890.665585	—	30436.587611	21635.183297	—	45652.719643	—	—	—	—	—
8	120121.212446	89861.312352	—	96946.44598	47849.849233	—	91915.972515	—	—	—	—	—
9	105942.531586	178876.690535	—	103848.223766	87565.248963	—	182948.698773	—	—	12373.5	—	16807.4
10	21.200884	20.864971	22.245619	20.758841	20.544966	21.116246	23.171215	3.24482	17.557405	0.098368	0.083639	0.035535
11	—	249875.639164	—	—	116615.448359	—	254221.414281	—	—	—	—	—
12	—	36928.321603	—	—	25765.916616	—	—	—	—	1128.02	—	9653.81
13	16.989145	16.824699	17.024381	16.969187	16.264646	16.289116	18.995765	1.168709	17.101061	0.059973	0.128092	0.073875
14	—	95894.416914	—	—	55972.2665	—	100907.636521	—	—	—	—	—
15	16.807833	16.713155	17.251369	16.922354	17.056758	17.063217	19.486674	0.994828	18.291201	0.066545	0.07243	0.072435
16	20.6055	20.527607	20.942258	20.47675	18.910981	19.913015	23.520856	3.639858	15.416263	0.092292	0.039808	0.032357
17	184056.301042	—	—	166913.895133	184620.172847	—	—	—	—	10842.8	—	—
18	22.177628	25.0118	159.328237	23.176467	25.839925	189.976448	27.005697	502.980908	106.019146	0.830984	1.28571	1.143
19	273492.273852	—	—	235780.516569	239713.780994	—	—	—	—	—	—	—
20	—	—	—	—	20326.769993	—	—	—	14.128244	5.21136	0.029817	0.027678
21	18.620333	18.167771	17.649685	18.200941	18.186755	18.02476	19.765213	—	237.896288	5.49234	2075.74	12.0054
22	30017.818487	40.266265	—	20230.000809	29.234852	16018.464315	43.828668	—	2025.721988	1.17849	51.4027	1.85201
23	—	152553.071717	—	—	90432.83243	—	—	—	—	—	—	—
24	165382.195993	241847.07017	—	152071.000286	131789.973055	—	—	—	—	—	—	—
25	—	—	—	—	97293.702811	—	—	—	24.069194	4.84413	0.069058	0.070298
26	149.103448	22.626461	978.851695	169.141322	24.277867	1072.542207	24.648778	281818.943958	98.841766	11.2667	27.7896	21.3565
27	37780.398963	21.183765	—	26696.136747	21.040494	15053.008481	23.003475	—	2929.873912	2.11929	—	3.36781
28	109.002642	17.498203	805.936563	130.893395	18.289007	906.815831	20.347545	—	125.426948	8.47757	16.9098	15.2035
29	16.18018	16.539943	16.831302	16.653144	15.516199	16.687271	18.497197	0.529552	17.228671	0.059729	0.080399	0.06606
30	18.680234	19.297598	65.42806	20.464174	21.527321	69.480097	22.771787	2279.899643	4377.451111	3.40626	94.503	5.84106
31	112584.230469	106892.487048	—	106350.368801	59820.569176	—	116731.921405	—	—	—	—	—
32	18.847318	19.001098	19.276024	18.561686	19.131745	19.343208	20.862294	1.637654	—	1927.02	—	—
33	15.877677	16.633968	15.662549	16.940131	15.656215	15.739041	19.051578	0.459259	16.449256	0.075777	0.121778	0.131984
34	23.833475	52.303108	116.363368	25.416274	52.173101	122.693402	41.858172	547.925213	106.06356	0.449711	1.11023	0.512011
35	15.244248	16.859991	17.273988	16.428828	15.502607	17.134687	18.956044	0.483225	10.024851	0.055403	0.048495	0.041678
36	—	10779.514788	—	—	7474.839078	—	—	—	11350.283337	342.1	—	853.776
37	194143.982081	—	—	183772.059796	153911.64927	—	—	—	—	—	—	—
38	21.850777	25.5672	434.369505	23.319772	26.536098	509.687074	27.391043	1216.057957	1632.068686	1.94115	32.1651	2.8346
39	16.423516	16.452341	17.551999	16.419512	17.4089	17.193939	19.85607	1.177408	8.748741	0.065532	0.026006	0.023087
40	16.777183	17.077404	18.00011	16.640976	16.291248	17.028004	20.153962	—	327.688167	6.02855	5257.02	18.6328
41	15.466287	16.464266	15.292817	15.355776	15.313759	15.403857	18.957689	0.839953	11.143646	0.045749	0.054575	0.040143
42	279656.987457	149758.509196	—	242456.032139	102462.012582	—	276128.94699	—	—	—	—	—
43	—	293629.571325	—	—	175758.39622	—	—	—	—	—	—	—
44	—	—	—	—	—	—	—	—	—	—	—	—
45	19.497227	19.351264	19.127426	19.122297	19.698612	19.091178	21.683916	5.392415	7303.509533	3130.66	—	4358.56
46	19.911364	20.799284	64.815077	21.196554	22.354869	69.300221	23.282352	2053.324053	3969.177388	3.30501	93.8687	7.05162
47	248.647304	27.265209	1234.019359	284.942522	28.325648	1335.919013	29.433656	99227.115311	117.777574	5.98298	17.0413	23.9457
48	—	—	—	—	—	—	—	—	—	725.899	—	1316.68
49	31660.015362	44.988977	—	20357.347037	30.873643	18288.888469	46.644426	—	2437.966819	2.08067	106.275	3.98303
50	—	—	—	—	169708.337508	—	—	—	—	—	—	—

Table 13: Results on *Random Lydia Case 08 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	—	—	—	—	202624.173201	—	—	—	25.582577	32.5515	0.067548	0.033464
2	—	—	—	—	—	—	—	—	—	—	—	—
3	18.689927	18.482602	19.206651	18.446774	18.401904	18.64912	19.907831	—	1004.314178	22.5193	35900.2	217.157
4	18.556394	19.312037	19.045033	17.92455	19.07326	19.28501	21.067634	1.790123	—	—	—	—
5	—	—	—	—	—	—	—	—	—	—	—	—
6	—	102664.739203	—	—	71304.804389	—	—	—	—	16434.9	—	—
7	35914.243109	48.799137	—	22568.16775	32.73514	—	51.666327	—	2649.604001	4.06637	—	15.3771
8	—	253.203577	—	262562.923917	434.751429	—	265.900157	—	11710.179313	135.957	—	1561.16
9	—	—	—	—	—	—	—	—	—	—	—	—
10	—	—	—	—	—	—	—	—	37.084158	42.7727	0.073089	0.033802
11	17.406469	17.566853	16.994471	17.297509	17.429424	16.974202	19.454931	1.854803	30.463553	0.07465	0.180101	0.198227
12	—	264.909548	—	—	64.963186	—	280.967496	—	12116.241854	4.07561	—	19.454
13	—	—	—	—	280052.81727	—	—	—	—	—	—	—
14	—	—	—	—	—	—	—	—	—	453.519	—	1076.92
15	427.875008	49.851753	3617.983631	373.420804	51.008008	3280.230652	52.889348	—	163.278704	17.9401	92.1141	108.042
16	22.056381	21.601903	21.402869	21.22219	22.003891	21.719739	24.221701	3.638146	15.804157	0.148472	0.051698	0.053838
17	—	—	—	—	—	—	—	—	—	—	—	—
18	22.875395	22.902142	23.04227	23.053399	22.64783	21.668687	25.651635	4.554339	20.026471	0.137049	0.039567	0.041123
19	—	—	—	—	276614.027061	—	—	—	—	—	—	—
20	—	127.679747	—	152969.208056	214.735242	—	135.707587	—	4870.939733	82.5162	—	621.849
21	18.063639	18.168886	17.9081	18.156082	17.669599	18.858741	21.200775	—	966.255249	19.8185	35824.9	86.7208
22	—	—	—	—	—	—	—	—	—	1368.6	—	2732.14
23	214.810305	24.377995	2674.531072	268.407055	25.129171	2897.765845	26.210734	—	398.36192	15.4819	84.0001	100.557
24	—	251.231309	—	260724.012333	434.448379	—	264.795461	—	9363.398194	135.721	—	1552.31
25	—	—	—	—	—	—	—	—	31.433031	62.9247	0.0336	0.037917
26	17.98733	18.778486	18.175267	18.286952	18.238803	17.757906	19.985848	—	224.443449	6.80783	1878.26	20.5224
27	—	—	—	—	—	—	—	—	—	—	—	—
28	24.102854	23.940298	23.729757	23.767862	23.543837	23.409177	25.537464	5.469836	18.634229	0.115413	0.084761	0.087624
29	—	—	—	—	210982.34043	—	—	—	—	—	—	—
30	17.200728	17.128344	17.348958	17.188536	16.985155	17.154416	19.307225	0.621151	39.666088	0.076217	0.175426	0.194824
31	18.316338	19.233153	17.844302	18.350554	18.850374	17.749246	21.080046	1.31361	—	1552.7	—	—
32	18.802679	17.251663	17.572239	19.068368	17.245141	17.435939	20.704047	—	1329.906145	28.2298	—	278.41
33	19.776704	18.940311	18.01242	19.382039	17.49408	19.067017	21.160141	6.98768	85680.051348	—	—	—
34	21.607727	19.833626	20.267552	21.318107	20.456751	21.087629	23.134381	3.696642	16.11936	0.149877	0.13496	0.045717
35	193.325717	18.833912	1079.573208	94.053949	18.662874	782.250947	20.653238	21940.166186	56.317516	2.46479	3.92686	5.65143
36	19.566776	17.302546	18.761146	19.623483	17.695169	18.822897	20.960719	1.365216	—	2998.2	—	—
37	17.47695	17.050018	17.65712	17.512082	16.646572	17.535998	18.79189	0.507129	32.539128	0.072557	0.150065	0.244685
38	—	—	—	—	—	—	—	—	44582.900961	1520.16	—	—
39	—	—	—	—	—	—	—	—	—	7792.92	—	—
40	23.884394	23.050333	22.880375	23.04253	22.874374	22.828246	24.934487	4.420263	20.843338	0.101713	0.078704	0.085425
41	121808.876199	124.673132	—	43234.176589	203.262006	—	128.721111	—	3261.413109	36.3502	16694.1	231.517
42	18.633163	17.341	18.353679	18.430417	17.528151	18.419073	19.795934	—	909.931605	17.4468	32434.3	196.46
43	22.614027	30.004244	1885.370931	24.615876	31.84602	2163.667734	33.064699	5222.225651	9541.85721	7.62647	203.987	28.6809
44	—	—	—	—	269394.906351	—	—	—	19.90659	35.98	0.04042	0.048914
45	168.083687	26.653953	2355.232076	221.061957	28.076017	2611.3152	29.997087	—	134.418919	3.88001	5.4512	7.00168
46	19.025471	17.064875	17.677436	18.553497	16.69701	18.573641	20.507404	—	958.965537	17.1833	38537.4	187.228
47	16.767765	15.811001	17.559552	17.200257	15.951787	17.317494	19.166961	1.078338	30.813243	0.067606	0.1624	0.201867
48	—	—	—	—	—	—	—	—	—	—	—	—
49	18.029356	17.985733	16.849153	18.11485	17.858815	16.973439	20.561123	1.88034	11.12712	0.104645	0.080152	0.031448
50	262407.892568	—	—	282131.53332	296839.320663	—	—	—	—	—	—	—

Table 14: Results on *Random Lydia Case 09 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	200.296323	201.567275	203.168077	212.315198	211.333889	215.085183	214.71067	—	3765.572701	97.1977	—	880.774
2	—	—	—	—	—	—	—	—	—	—	—	—
3	20.147491	19.405688	20.195466	19.662369	20.110029	20.140943	21.846456	12.447289	—	—	—	—
4	996.878624	44.910837	13744.3846	1099.670634	46.319641	12883.353099	48.160912	—	933.194199	95.7317	713.011	834.38
5	—	—	—	—	—	—	—	—	—	—	—	—
6	19.585303	20.233692	19.673646	19.668884	19.37875	19.736536	21.890737	11.056458	—	—	—	—
7	—	—	—	—	—	—	—	—	—	—	—	—
8	534.117976	19.59044	6090.845023	406.786223	20.614786	5752.704324	21.771562	—	496.888634	96.56	514.965	608.349
9	—	278.570826	—	—	461.805236	—	288.673375	—	31019.403569	176.118	—	867.472
10	—	—	—	—	—	—	—	—	—	—	—	—
11	—	—	—	—	—	—	—	—	—	—	—	—
12	17.505689	17.298959	17.416763	17.737131	17.110917	17.481125	20.021202	3.166652	61.725406	0.112209	0.447661	0.585107
13	—	53.56902	—	—	33.707265	—	56.982615	—	117605.335016	43.5619	—	172.915
14	—	—	—	—	—	—	—	—	—	—	—	—
15	585.116591	20.224199	8285.270866	413.593697	20.641122	7365.119343	21.077476	—	457.507922	319.046	—	—
16	—	—	—	—	—	—	—	—	—	—	—	—
17	1891.30431	41.835165	14414.404324	2060.02855	43.648632	15363.806665	43.222086	—	954.950878	113.093	802.317	884.28
18	—	—	—	—	—	—	—	—	—	—	—	—
19	28.25844	28.170182	27.749755	27.96281	28.144196	26.161355	30.919007	9.480531	24.080528	0.184404	0.055062	0.042064
20	—	—	—	—	—	—	—	—	304.409219	331.094	0.032945	0.036227
21	—	—	—	—	—	—	—	—	23.66599	339.687	0.037214	0.039295
22	18.859243	18.522718	18.850804	18.777523	18.494124	18.576399	20.823638	2.285719	9.58564	0.124427	0.027258	0.032123
23	—	—	—	—	—	—	—	—	53.234065	400.021	0.048801	0.051911
24	28.554814	28.313811	28.691009	28.791152	28.162286	27.138076	30.939031	6.301981	23.056391	0.239977	0.057292	0.065249
25	19.031579	19.310545	19.865352	19.050093	17.786393	17.965267	21.807561	1.791023	—	—	—	—
26	19.246743	19.813456	19.906902	19.714937	19.447442	18.30384	21.600346	13.655258	—	—	—	—
27	—	—	—	—	—	—	—	—	22.157384	118.031	0.071023	0.078694
28	—	—	—	—	—	—	—	—	—	—	—	—
29	22.449315	21.987618	22.034307	21.334802	21.942872	21.886558	25.115722	6.938389	19.870879	0.192848	0.04919	0.060071
30	—	—	—	—	—	—	—	—	—	—	—	—
31	—	—	—	—	—	—	—	—	—	—	—	—
32	—	24.101913	—	—	23.606858	—	25.605592	—	172110.597835	47.0083	—	180.165
33	—	—	—	—	—	—	—	—	—	—	—	—
34	28.186903	28.305976	28.754304	27.371607	27.312023	27.950146	30.38891	6.334617	23.750319	0.161832	0.077504	0.044852
35	19.590999	19.320211	19.716771	17.665698	18.970661	18.212797	22.366145	4.759862	24572.636922	—	—	—
36	—	—	—	—	—	—	—	—	—	—	—	—
37	—	—	—	—	—	—	—	—	—	—	—	—
38	—	—	—	—	—	—	—	—	—	—	—	—
39	19.398852	19.715057	19.769205	19.340182	19.507698	18.436303	22.102144	27.188393	—	—	—	—
40	—	—	—	—	—	—	—	—	—	—	—	—
41	—	—	—	—	—	—	—	—	—	—	—	—
42	22.478374	23.033107	22.782828	22.048596	22.49234	22.597907	24.271406	4.383399	19.632279	0.219921	0.09619	0.097988
43	20.635724	19.882713	21.058818	20.240772	18.730647	18.643832	22.801462	10.594384	—	—	—	—
44	—	195541.175723	—	—	164335.468253	—	—	—	—	—	—	—
45	29.1681	28.874478	28.119487	27.803505	27.504798	27.624774	30.376049	5.178641	20.292778	0.160278	0.079334	0.079631
46	20.120603	19.167374	19.337507	17.671985	19.254439	18.223237	21.474227	11.945624	—	—	—	—
47	—	—	—	—	—	—	—	—	—	—	—	—
48	18.682479	18.755441	18.338184	18.113817	18.177409	18.401453	20.14213	—	1953.194231	34.9219	—	195.364
49	—	—	—	—	—	—	—	—	—	—	—	—
50	28.971457	28.671703	28.522456	28.13169	28.164525	28.276661	31.169693	9.889272	25.313422	0.182945	0.089335	0.045606

Table 15: Results on *Random Lydia Case 10 50*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	—	—	—	—	—	—	—	—	28.782413	60.3576	0.032948	0.037117
2	—	—	—	—	—	—	—	—	—	—	—	—
3	17.575102	17.352167	18.173289	17.371955	17.395569	17.545897	18.723553	1.080734	140.346973	0.200886	0.597075	0.572767
4	—	574.387805	—	—	981.787471	—	587.057097	—	157430.510276	860.009	—	1093.0
5	—	—	—	—	—	—	—	—	—	—	—	—
6	17.371103	17.750082	17.536003	17.546889	17.228279	17.089149	18.940858	1.191698	141.895564	0.197331	0.64155	0.526382
7	—	297.350363	—	—	565.279784	—	303.154461	—	83451.112641	1111.48	—	5069.74
8	—	—	—	—	—	—	—	—	83.362532	488.51	0.047216	0.03603
9	—	—	—	—	—	—	—	—	—	—	—	—
10	—	—	—	—	—	—	—	—	40.052831	318.29	0.075553	0.073298
11	—	—	—	—	—	—	—	—	—	—	—	—
12	—	1123.95505	—	—	1626.926109	—	1212.909402	—	—	3797.13	—	4292.7
13	19.307176	19.935293	19.468371	19.577817	19.89629	19.67229	21.614557	—	15128.326275	408.815	—	3086.72
14	15.811408	16.624685	17.482837	17.035327	16.789336	16.855076	19.581005	1.075849	61.180518	0.108772	0.393716	0.290431
15	—	—	—	—	—	—	—	—	—	—	—	—
16	—	—	—	—	—	—	—	—	—	—	—	—
17	29.30735	28.891086	29.233792	28.217818	29.205045	28.277628	30.944918	9.76887	23.631755	0.36434	0.090895	0.084076
18	—	—	—	—	—	—	—	—	—	—	—	—
19	—	—	—	—	—	—	—	—	—	—	—	—
20	—	—	—	—	204447.016053	—	—	—	—	—	—	—
21	18.978805	18.65948	18.837451	19.749085	19.421608	19.001093	22.077104	2.944238	13.788441	0.220771	0.075605	0.070925
22	—	—	—	—	—	—	—	—	—	—	—	—
23	—	—	—	—	—	—	—	—	—	—	—	—
24	—	—	—	—	—	—	—	—	—	—	—	—
25	17.664698	17.350628	17.699025	17.754208	17.580507	17.478857	19.709357	3.24919	134.277958	0.213936	0.682873	0.57274
26	—	—	—	—	—	—	—	—	—	—	—	—
27	24.372136	34.904099	672.729441	26.048133	37.891423	860.133638	37.678796	1111.262669	1553.191567	7.20715	38.7803	20.181
28	26.699376	27.639418	28.873281	28.219889	28.538878	28.073293	29.626727	15.57836	28.346627	0.404512	0.061239	0.052158
29	—	448.395892	—	—	305.627611	—	461.884847	—	3966.068165	15.3239	—	5.54832
30	23.244622	36.147881	1875.881298	34.498614	37.227489	2214.888375	37.617141	3530.297388	9380.365713	34.1356	—	77.4551
31	—	—	—	—	—	—	—	—	—	—	—	—
32	—	—	—	—	—	—	—	—	—	—	—	—
33	17.644231	17.478486	17.28789	18.434666	17.967924	17.541657	18.737701	0.62762	136.946041	0.194661	0.660158	0.613405
34	24.387423	87.029666	2528.426838	27.372095	90.742544	2743.069545	42.39416	2609.39982	2625.937534	9.70728	49.1341	5.2198
35	—	—	—	—	—	—	—	—	—	—	—	—
36	29.159523	29.940538	29.882961	29.138067	29.477853	29.222763	31.853127	11.379259	28.369612	0.321665	0.052205	0.044892
37	—	—	—	—	—	—	—	—	—	—	—	—
38	—	—	—	—	—	—	—	—	—	—	—	—
39	28.799718	28.241571	28.454633	28.426142	28.372775	27.383608	30.347658	4.752875	22.102583	0.292842	0.110732	0.10539
40	—	—	—	—	—	—	—	—	—	—	—	—
41	26.032707	34.998464	1643.707461	36.937454	36.946626	1970.217552	36.719967	5361.711451	9763.86672	43.0906	—	13.4229
42	—	—	—	—	—	—	—	—	39.595451	309.962	0.080362	0.032016
43	19.525821	19.194809	19.173425	19.089618	19.224351	19.192651	21.406796	3.642171	15.30179	0.212786	0.026099	0.024779
44	580.790637	18.533042	14586.592202	376.334587	20.902793	12032.695202	21.431082	—	607.365575	218.259	—	978.357
45	3165.431081	67.384064	33401.568292	1823.355478	68.233522	28171.919469	73.998536	—	165.097198	40.1551	186.18	172.881
46	—	—	—	—	—	—	—	—	—	—	—	—
47	—	—	—	—	—	—	—	—	—	—	—	—
48	—	—	—	—	—	—	—	—	—	—	—	—
49	—	—	—	—	—	—	—	—	—	—	—	—
50	—	296.521874	—	—	623.837349	—	302.797427	—	75010.002967	473.546	—	3315.42

Table 16: Results on *Random Syft 01, 0-100*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	17.838381	18.010678	19.426528	18.434495	18.272431	19.153742	25.453386	1.480109	17.149557	0.06058	0.097848	0.041566
2	15.555838	16.936421	16.358218	16.154483	15.942102	17.039143	18.64216	0.433352	17.302952	0.21697	0.087593	0.08387
3	244.892236	252.21823	2774.826025	264.237455	273.860434	2827.775795	256.829823	7029.127958	10.39073	—	0.022478	0.018531
4	17.890429	18.123882	18.386884	18.493321	18.205205	18.900738	19.865162	0.95174	11.366788	0.043558	0.055002	0.040489
5	2304.336286	2557.382003	—	2695.759816	554.392568	267737.691448	2574.013279	—	94.161289	205.777	3.52906	2.99827
6	19.385058	19.853257	18.833662	19.287574	19.506592	19.331106	21.038914	1.37929	14.260233	0.048421	0.048696	0.048428
7	16.689495	17.907299	18.143165	18.205655	17.339884	17.51406	19.608336	0.808236	73.956006	35.799	5.08427	4.73567
8	15.624526	15.633984	16.042865	16.168236	16.1247	16.46547	18.422168	0.374489	16.366283	0.182094	0.627471	0.567531
9	17.984099	17.476691	16.748291	17.076929	17.97599	17.106557	19.494134	0.855414	10.610905	0.042498	0.036051	0.034236
10	17.685524	18.434985	18.692184	18.5878	17.938472	18.103472	20.628276	1.252084	14.802912	0.040851	0.042319	0.037315
11	389.619713	419.762999	60576.314704	420.830747	130.066422	14209.226413	423.536608	—	25.077844	10.5695	0.421491	0.411797
12	19.070878	18.864848	18.31963	19.247821	18.830747	19.113381	20.321897	1.204464	13.288654	0.043428	0.043071	0.045563
13	15.887511	16.77877	17.107754	16.62774	16.516723	16.034154	18.624028	0.498387	17.666278	0.213673	0.102817	0.107286
14	339.821205	364.239667	44226.770913	358.522842	122.971479	8039.982239	367.73303	—	27.976605	3.80797	0.503232	0.467393
15	162.272237	174.798238	11789.731216	177.251406	74.224155	2729.808089	177.15959	—	20.627745	1.75585	0.228625	0.173933
16	1509.966477	1566.454966	27261.589853	1682.550258	1717.016302	27560.277917	1581.280398	103131.79842	12.818765	—	0.019117	0.018583
17	142.819737	151.45346	8151.451692	155.414205	71.099953	2011.102901	153.354157	—	22.114329	0.882753	0.223118	0.18435
18	838.29836	894.617486	221379.839928	965.46771	241.680447	41075.694593	898.638368	—	38.109586	156.856	1.01648	0.963573
19	17.270588	16.967485	17.321624	17.511575	17.354024	16.7567	18.494921	0.569946	17.003998	0.208543	0.088764	0.10037
20	386.317438	418.668603	60513.798071	417.569434	130.361392	14205.189526	421.642649	—	25.719567	10.1365	0.412505	0.374022
21	17.747976	18.208437	17.610393	18.272474	18.511557	18.094666	19.536769	0.918224	72.920621	36.3237	4.99377	5.60264
22	15.316318	16.186033	16.229343	17.030148	17.366439	16.348987	18.304633	0.345069	13.934593	0.078271	0.13289	0.128495
23	603.218039	627.781188	9601.708496	693.869988	734.398143	9851.581664	633.240728	53268.810636	11.831575	—	0.019162	0.018439
24	184.382973	228.445676	11820.075855	197.647765	81.761266	2477.502089	230.802229	—	20.082156	0.250827	0.239595	0.20003
25	1088.027511	1345.104983	—	1297.532657	283.541843	50718.754348	1343.744655	—	37.39476	1.57995	1.36921	1.33312
26	19.106695	18.892473	19.907648	19.044579	19.745137	19.869472	20.909851	1.534108	16.588936	0.042317	0.038912	0.043676
27	439.359316	549.218957	61480.890203	535.928432	144.645399	12165.196185	555.43626	—	25.370698	0.401692	0.384919	0.359069
28	65.34019	68.060989	1446.174353	77.126	47.911932	437.316258	71.310846	—	18.448045	0.301191	0.11007	0.090017
29	19.510086	19.480615	19.626307	20.114163	20.207296	19.883162	21.058823	1.517884	14.749464	0.04068	0.039173	0.037379
30	17.796464	17.141718	18.067132	18.230111	18.835673	18.470463	20.229483	1.077217	11.108248	0.041724	0.041413	0.040371
31	834.704614	892.352369	219925.925031	958.480101	240.512728	40921.902029	900.103927	—	37.909149	156.239	1.02738	0.973671
32	18.853431	18.441649	18.816021	18.324635	18.355627	18.93115	19.582102	1.121247	10.856368	0.046129	0.046183	0.041159
33	18.51626	19.210881	18.349824	19.086112	19.282884	19.357208	20.895474	1.513182	15.973199	0.041817	0.038651	0.041744
34	72.833878	76.776238	2097.753697	84.571508	48.041133	50.076369	79.561967	—	15.674648	0.508805	0.10821	0.076669
35	19.516778	18.712961	18.999099	19.222748	19.384858	19.205386	20.308106	1.38979	12.717289	0.043439	0.041494	0.039623
36	2082.916478	2203.328206	—	2448.639765	527.584372	188829.191301	2215.713089	—	63.698237	96.7323	4.55131	4.25849
37	19.461929	19.4699	19.806604	20.128936	19.791491	19.925405	21.238381	1.58849	15.597709	0.041249	0.038553	0.040421
38	19.274954	18.53945	19.018333	19.367378	19.018674	18.803317	21.387779	1.672004	19.34519	0.042722	0.044483	0.042277
39	604.462492	630.736384	9661.670611	693.966054	735.466325	9784.947528	632.084144	60819.085273	11.673504	—	0.023862	0.019667
40	18.561429	18.489886	18.236152	18.464858	18.564716	18.605955	19.995922	1.000484	10.700197	0.043309	0.046731	0.03962
41	445.460579	550.610109	61408.85439	535.94689	144.539823	12163.77802	554.318357	—	24.984857	0.404265	0.401735	0.35505
42	17.86007	17.721834	17.624715	17.223047	17.144118	17.04507	19.088906	0.735273	26.381654	2.94813	0.333768	0.329457
43	338.704793	362.050383	42850.212669	358.159441	122.117459	8075.993002	365.77127	—	27.308999	4.08317	0.500809	0.42975
44	19.862606	19.481509	19.116644	19.429932	19.585253	19.875229	20.796584	1.505084	16.363736	0.049731	0.039396	0.037756
45	16.050377	16.30501	16.773092	17.196538	17.557337	17.54545	19.086706	0.44466	31.430007	11.7493	44.6026	41.2842
46	18.95409	18.887318	18.211898	19.110623	18.677201	18.93475	20.459382	1.230969	13.449608	0.053191	0.041114	0.042503
47	17.329796	17.588465	16.165224	17.781723	17.341248	17.693352	19.121568	0.752094	40.004619	8.86861	0.699018	0.635636
48	16.502577	17.549346	17.206256	17.193872	18.173447	18.720152	19.948513	0.989553	11.253373	0.056921	0.043817	0.039984
49	183.970155	227.830665	11713.627335	196.482371	81.388342	2469.984018	229.619755	—	20.147378	0.260683	0.25392	0.233146
50	19.082125	18.9206	18.521306	19.159651	19.112681	19.330766	20.721196	1.358353	14.450093	0.048504	0.044314	0.045736
51	16.538293	17.257886	16.406874	17.783555	18.020055	17.469827	19.15471	0.82261	71.808384	35.4888	4.98909	5.00492
52	17.312163	16.306766	15.854163	17.608147	17.153367	16.688082	19.074556	0.499486	30.719126	11.6318	44.1242	41.7962
53	17.110113	15.910061	15.543221	17.343677	16.879322	16.289783	18.895444	0.543965	20.937185	0.665605	0.156675	0.142058
54	183.72581	227.190041	11730.799413	197.584271	82.114315	2479.801962	229.936294	—	20.480717	0.26582	0.246579	0.192936
55	246.399503	254.620197	2765.464936	265.754738	274.804509	2812.997654	257.039386	7670.893788	9.965514	—	0.019834	0.019059
56	73.822753	84.853201	2102.386923	84.824374	49.913332	50.415233	79.55685	—	17.025365	0.520329	0.102664	0.076228
57	839.843242	898.444533	220098.657839	960.799852	244.150235	40901.131864	899.936556	—	38.897168	160.437	1.02752	0.940999
58	143.203853	152.306944	8137.994862	156.47147	71.193879	2011.781255	153.649471	—	21.452164	0.919024	0.219583	0.199535
59	18.257701	18.506917	18.454695	18.036329	18.319724	18.507305	19.575336	1.013787	9.699644	0.038116	0.03706	0.035189
60	18.120745	18.169974	18.223305	17.531822	18.193501	17.825478	19.699619	0.929932	10.428169	0.037448	0.036146	0.03496
61	18.910568	19.246255	19.798143	19.928485	20.21575	19.526309	21.427354	1.627314	17.904303	0.04214	0.03965	0.040634
62	1505.899065	1565.904292	27180.572077	1680.162522	1725.600163	27552.374465	1570.65331	89501.220672	13.684823	—	0.0203	0.019172
63	50.167838	50.997964	148.79575	52.846748	54.396983	152.980537	52.906594	733.876396	7.743211	—	0.02767	0.018136
64	19.147819	19.097725	18.813736	18.732769	18.982399	18.93032	20.633226	1.30917	13.604488	0.047091	0.043339	0.045745
65	2620.841221	167566.582198	—	3028.11548	1880.302314	—	5048.219751	—	85.099962	6.49458	8.61049	5.42297
66	18.994852	19.104061	18.948748	19.068567	18.972374	19.473389	19.981251	1.226957	12.64131	0.053758	0.048452	0.041614
67	18.393297	18.058215	18.129538	17.610861	18.071889	17.873825	19.031524	0.807879	72.120237	36.137	5.09362	4.76134
68	961.745988	1027.746258	—	1088.914084	255.982329	61833.197998	1035.550623	—	39.158085	46.6199	0.68693	0.640986
69	18.30881	18.052403	18.013414	18.289102	17.627037	17.977935	19.958002	0.967749	9.92455	0.036971	0.036612	0.042474
70	17.916782	16.500189	16.507784	17.905199	16.806537	18.269319	19.456554	0.806608	71.834252	35.899	5.32392	4.86028
71	19.046806	17.802542	18.302599	18.808982	19.222108	19.309349	20.739749	1.346071	16.225406	0.041273	0.04123	0.042496
72	1502.339493	1564.010598	27314.216053	1667.897116	1725.231331	27563.656516	1568.589682	98144.675877	13.186243	—	0.01958	0.018976
73	606.297548	627.991612	9612.548223	693.454067	735.187176	9888.086702	634.277586	60587.899753	11.310017	—	0.023115	0.018529
74	17.655822	17.504637	17.444252	17.502487	18.092495	17.869764	19.044753	0.525863	30.375586	11.7154	46.3648	42.6693
75	184.522382	227.593475	11803.12315	196.25699	82.423162	2477.132978	231.544012	—	20.615526	0.261122	0.242487	0.207454
76	19.463433	19.48375	19.221506	19.595326	19.839377	19.0083	20.769505	1.525629	16.514527	0.048342	0.044215	0.037979
77	50.141944	49.994043	148.296536	53.31466	54.153682	153.796406	52.398876	747.516586	7.671221	—	0.020452	0.022182
78	2078.973517	2217.587265	—	2447.113452	527.898237	188859.364436	2221.348443	—	66.347634	96.1493	4.78902	3.93807
79	19.091119	19.034207	19.05833	19.088223	19.400005	19.427604	20.547832	1.3619	14.337009	0.047325	0.03983	0.039225
80	102.561338	106.062828	419.58354	119.290861	122.961028	468.459137	110.163705	1183.393015	8.976944	—	0.019682	0.019126
81	17.405454	16.489675	17.525211	17.467677	17.387875	17.288714	18.73548	0.439646	15.283343	0.183134	0.618921	0.573761
82	837.853067	892.040404	219590.691425	964.613796	241.934844	40910.250965	901.849547	—	39.34125	155.688	1.02472	0.9828
83	17.615493	17.199498	17.403029	17.590183	17.702883	17.461639	18.642304	0.680928	27.330104	2.92566	0.320324	0.29777
84	16.301263	15.645244	16.600874	15.99343	17.288325	17.349964	18.876741	0.427344	15.539014	0.188435	0.664727	0.611289
85	18.161215	18.255417	18.157552	18.322022	18.886528	19.512379	20.937977	1.382395	16.10217	0.043902	0.039958	0.056035
86	19.457944	18.05302	19.000915	18.971998	19.698684	19.791431	20.935581	1.382326	15.086646	0.043321	0.039027	0.040592
87	1506.606513	1555.174862	27527.160239	1679.4498	1723.15849	27563.307637	1565.448034	89213.142732	12.905669	—	0.019657	0.01938
88	19.808824	19.181858	19.923596	19.450722	19.421092	19.643954	21.443542	1.512323	15.462491	0.042461	0.041897	0.037847
89	604.978695	627.629519	9651.121389	690.163286	737.020465	9802.548125	633.111709	57348.48895	11.456716	—	0.020084	0.019037
90	82.677023	100.053102	2245.079272	93.529206	53.657448	527.130957	101.720539	—	16.556631	0.161916	0.122032	0.093962
91	338.843202	362.55532	43035.217554	359.000421	121.700379	8051.625901	368.199601	—	27.993134	4.09917	0.530752	0.453141
92	104.373122	107.174957	415.909708	118.962367	135.158076	469.384344	109.52915	1278.514058	8.922729	—	0.019022	0.022135
93	19.548517	20.217582	20.132645	20.112914	20.301001	20.488955	21.116389	1.751711	18.265656	0.040947	0.04187	0.03708
94	18.761154	18.208567	17.80263	19.75555	19.270346	19.046696	20.442227	1.305112	14.109473	0.041966	0.040265	0.039094
95	15.770447	16.177787	15.473282	17.110182	17.179447	16.543033	18.190152	0.414422	12.951078	0.072413	0.141479	0.126151
96	602.710772	636.027282	9621.318732	692.283353	733.020054	9784.595758	632.597816	59057.55237	12.283027	—	0.01879	0.018215
97	1510.081798	1576.592584	27252.38667	1682.321784	1727.332303	27597.779842	1569.984876	95869.609498	13.114178	—	0.018764	0.018383
98	18.445672	18.643773	19.098869	18.470905	19.002688	18.833405	20.313615	1.139242	12.335054	0.047052	0.042133	0.042254
99	17.976526	18.02166	18.304801	18.088995	19.862373	19.217505	20.705746	1.286753	13.239812	0.047424	0.040735	0.042211
100	49.081453	49.36695	147.572494	51.854593	54.235075	153.448182	52.920622	739.336432	7.872234	—	0.019015	0.017899

Table 17: Results on *Random Syft 01, 100-200*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
101	17.049827	17.433849	18.03419	18.127264	18.324177	18.168057	19.728269	0.986675	10.005363	0.036104	0.035689	0.037083
102	17.298718	18.766137	17.739139	18.916295	19.015619	18.320205	20.296427	1.240988	14.055802	0.042816	0.043886	0.040517
103	2619.760211	165036.284525	—	3028.209367	1865.938059	—	5049.075983	—	83.840387	6.97416	5.87934	5.43837
104	73.593391	79.033897	2107.287044	85.097798	50.440105	51.304254	79.462928	—	16.982128	0.517877	0.09886	0.08593
105	18.612943	19.336494	18.767024	19.363399	19.432526	19.680179	20.691308	1.296773	14.172291	0.046593	0.041691	0.042389
106	65.523752	68.227214	1445.416127	75.811361	47.489396	432.719049	70.144778	—	18.532368	0.284118	0.118855	0.091792
107	2312.880279	2561.363222	—	2707.34949	561.11436	267230.115431	2567.199951	—	91.060229	204.388	3.63981	2.78289
108	19.228815	19.273509	19.015467	18.813943	19.040968	18.982281	20.168209	1.325333	14.2508	0.040745	0.042506	0.038822
109	18.191138	17.171185	17.944441	17.861194	17.245369	18.323573	19.687574	0.985249	10.688592	0.044652	0.043006	0.041547
110	17.075558	16.153034	16.125885	16.96995	15.831912	17.237933	18.985689	0.651945	26.424967	2.88086	0.328286	0.316617
111	16.475236	15.851799	15.937535	16.144981	16.828532	17.193215	18.395748	0.470689	17.348462	0.20178	0.087101	0.096783
112	961.805872	1030.509355	—	1087.279899	256.139186	61774.304495	1036.706261	—	38.495731	48.2833	0.667742	0.60487
113	836.393634	897.450764	219515.861343	973.172627	242.547964	41010.22362	901.911416	—	38.093461	165.723	1.05906	1.00214
114	17.903042	17.376601	17.524113	18.065467	17.334913	17.915165	19.152192	0.582949	55.164097	81.0875	496.651	431.563
115	17.336133	18.46162	17.12966	17.965182	18.220361	18.703912	19.469365	0.912812	11.405093	0.039521	0.037495	0.035331
116	17.926726	18.697924	17.557031	17.982384	17.588386	18.796433	20.268995	1.007876	11.634695	0.042792	0.045134	0.039451
117	81.993654	99.75528	2247.15703	94.154393	53.518616	526.68507	101.215964	—	17.034054	0.136738	0.136725	0.094062
118	17.904476	18.072066	18.60132	17.92315	17.758764	18.014484	18.892824	0.892585	72.128759	36.1074	5.12779	4.69636
119	18.546549	18.173002	18.975551	17.696253	18.082317	18.686461	20.671183	1.32996	12.978533	0.050307	0.041817	0.044363
120	2616.430744	165659.558079	—	3032.682309	1864.786895	—	5048.086822	—	84.586588	6.57214	6.27278	5.50561
121	143.162109	152.412643	8163.639682	154.882487	70.7082	2010.470887	154.221092	—	21.789841	0.902902	0.221344	0.186799
122	162.83131	174.764351	11794.062719	175.997141	74.491817	2716.81782	177.26857	—	20.237198	1.77054	0.227776	0.165493
123	50.554178	51.771589	149.785442	52.698737	54.196897	153.818662	53.431863	750.945521	7.749675	—	0.019322	0.019137
124	141.638446	151.524804	8160.402169	154.722122	70.978785	2012.356471	153.274867	—	21.524826	0.877752	0.212744	0.166639
125	142.250875	151.505442	8135.390537	155.731783	71.63575	2008.169135	153.721356	—	21.474958	0.904654	0.253286	0.168762
126	18.416535	18.716484	17.95453	18.092977	19.01304	18.527257	20.191816	1.178828	12.828639	0.045443	0.042092	0.039808
127	1501.18688	1557.830044	27243.768001	1689.245414	1724.667831	27646.003929	1570.038191	96964.366333	13.095918	—	0.01923	0.020617
128	17.246936	17.251308	16.70335	17.281655	17.408798	17.154768	18.846095	0.464704	19.496119	0.675386	4.13573	3.95502
129	142.126394	150.086775	8214.800724	155.180186	70.99405	2010.861762	153.902913	—	21.788289	0.938474	0.23945	0.185604
130	18.921524	18.761039	18.931694	19.372089	19.568875	19.48001	20.656468	1.521507	16.554805	0.041834	0.042671	0.044902
131	17.30866	18.444962	17.229147	18.021001	18.326418	18.560179	20.107929	1.056809	12.109954	0.047175	0.041227	0.042688
132	15.808345	16.948881	15.894809	17.049724	17.120662	17.223184	18.437962	0.429364	19.137191	0.705644	4.03995	3.95124
133	102.415604	105.41542	414.020237	118.778795	121.366985	468.856094	108.640728	1230.855029	8.844848	—	0.019673	0.019613
134	16.473551	16.100327	16.999268	16.907759	16.951787	16.987527	18.588778	0.41822	13.278457	0.072454	0.126165	0.126844
135	18.26968	17.835976	18.050291	19.648198	19.243136	18.861713	21.112436	1.47577	15.012387	0.046955	0.05269	0.044336
136	1090.193084	1336.835469	—	1300.957231	283.772941	51055.869399	1349.533714	—	37.438313	1.525	1.3733	1.33393
137	388.883085	419.860456	61299.162265	419.146507	130.511058	14177.342512	420.775537	—	24.538467	11.0081	0.439017	0.389265
138	17.783078	17.940587	18.284163	18.037334	17.891922	17.913619	19.531929	0.844803	39.02818	8.71848	0.69163	0.657243
139	2072.199444	2209.599996	—	2439.312296	527.381485	188965.360267	2217.582599	—	65.272774	100.966	4.95618	4.33511
140	73.357582	78.066923	2109.85134	85.859598	49.04168	51.184071	79.189733	—	16.641512	0.523635	0.124714	0.086369
141	142.994988	151.095591	8121.305127	155.754349	70.624344	2013.047596	153.414963	—	21.759242	0.927517	0.241768	0.194222
142	16.876042	16.603056	16.447427	17.158831	16.994469	17.122958	18.957769	0.425541	13.532459	0.081902	0.150109	0.12083
143	161.810373	173.263724	11840.219764	177.345836	73.845034	2714.31589	176.401874	—	20.805143	2.12439	0.234915	0.190368
144	17.40213	16.889225	17.400468	17.18751	17.066334	17.368074	18.900851	0.490555	15.179824	0.187329	0.652716	0.59615
145	17.852346	17.394884	18.246801	18.712281	18.169162	17.854484	20.293833	1.131709	12.933088	0.046933	0.04213	0.039902
146	72.624244	76.152939	2104.101698	84.029182	47.912493	49.47727	79.602207	—	16.943957	0.5174	0.103704	0.07983
147	18.689413	18.762857	18.348438	19.192667	18.914205	18.779789	20.615641	1.321913	13.098592	0.045356	0.042463	0.039116
148	81.525827	98.67698	2258.803304	93.408889	54.147093	526.082397	101.607884	—	16.856412	0.13624	0.144917	0.090797
149	16.693482	17.169654	17.300941	17.413589	17.335327	17.270656	19.086355	0.642621	20.425433	0.669621	0.194365	0.146484
150	160.668546	196.770013	11787.07908	176.270706	74.91965	2725.03771	176.697807	—	20.190396	1.79935	0.239056	0.19918
151	66.251373	68.341872	1439.551773	77.902299	47.740292	434.973714	70.121649	—	18.539992	0.283094	0.125074	0.09196
152	1508.308007	1564.24334	27336.69078	1686.352648	1725.248978	27586.158247	1579.542306	99055.161097	13.580415	—	0.020858	0.018186
153	104.637981	107.177594	415.073675	118.847985	121.736216	469.221888	108.281701	1213.421391	8.483275	—	0.021798	0.017956
154	339.570795	363.274258	42922.429458	359.292667	123.374049	8063.546408	364.452313	—	27.289818	3.97908	0.501185	0.424913
155	105.042617	107.521001	415.906355	119.158334	121.876736	470.729234	108.913491	1316.542701	8.610729	—	0.028584	0.022264
156	17.933981	18.064185	18.09483	17.930192	17.756939	18.123562	19.109874	0.784609	39.981719	8.6912	0.711645	0.643574
157	836.969657	898.515052	219790.26635	961.958593	242.30628	40921.218721	899.93889	—	39.772038	163.843	1.16158	0.965945
158	338.704365	364.058626	42964.730095	361.28814	122.270944	8057.530878	366.939687	—	27.272456	3.91956	0.502449	0.44223
159	17.516647	17.419048	17.644756	17.542356	17.249692	17.556693	19.17458	0.514601	20.707745	0.689205	4.11512	4.03541
160	19.109855	17.747136	19.142663	18.900899	19.18449	19.056741	20.444737	1.29768	13.084834	0.044615	0.043512	0.042227
161	1087.403229	1339.3061	—	1294.463932	282.830294	51504.275299	1341.651858	—	37.998088	1.52365	1.38804	1.34081
162	20.114953	19.245995	19.738411	19.688225	19.71485	19.941714	21.280325	4.884849	20.247985	0.041138	0.053858	0.039091
163	18.470795	16.916333	17.697835	18.592915	18.171507	18.523736	19.930127	1.549751	10.561949	0.042247	0.042154	0.040625
164	2303.107059	2569.830704	—	2698.231928	555.546033	266530.757214	2590.474596	—	92.460747	206.926	3.28639	2.76926
165	82.520475	100.469157	2298.673846	94.269132	53.174663	528.126835	101.826492	—	17.230081	0.145147	0.133508	0.1065
166	185.478937	229.261928	11717.08544	198.03131	81.448979	2469.893004	229.934383	—	20.777103	0.257728	0.237387	0.201827
167	50.448367	52.319976	148.182128	54.214064	53.747115	153.755788	52.741924	749.080697	8.182141	—	0.023724	0.018103
168	19.625469	19.240399	19.382382	19.342213	19.30636	19.78001	20.440304	1.916867	16.592541	0.043392	0.042177	0.037985
169	16.707475	15.509862	15.632176	17.053583	16.65465	17.364209	18.666517	0.98569	17.461429	0.208465	0.112827	0.079871
170	340.129698	360.173194	42958.022029	359.883132	121.78242	8060.429527	364.840667	—	27.805213	4.17558	0.537934	0.461241
171	82.481795	99.69026	2239.959415	93.715477	53.975823	527.569547	101.157251	—	17.265727	0.136207	0.127679	0.096821
172	184.358928	227.507826	11712.999167	198.195237	83.249126	2469.801471	230.230958	—	20.678141	0.260429	0.233421	0.212675
173	837.394282	891.94248	220858.587341	962.596283	241.443375	40796.453667	902.478936	—	39.826348	155.483	0.997468	0.933004
174	19.412765	19.183451	20.091084	19.61017	19.232746	19.377023	21.053249	2.12853	16.55714	0.046326	0.039768	0.039402
175	50.479159	51.022524	148.508604	53.24126	53.829158	153.07597	52.788633	736.424247	7.325467	—	0.019316	0.022577
176	388.593269	437.939648	60426.285056	415.968985	129.587573	14274.914953	424.13897	—	25.528255	10.8248	0.450645	0.395375
177	66.286201	68.335665	1439.313654	76.384899	47.44872	441.979698	70.042264	—	18.63157	0.278638	0.110293	0.086084
178	50.464998	51.146232	149.800248	52.873137	53.740762	155.355589	52.726979	733.725795	7.976308	—	0.019225	0.018096
179	18.812763	18.821958	19.07792	18.277836	19.073556	19.314379	20.236734	1.105995	11.760035	0.047853	0.041439	0.040529
180	73.118469	76.556849	2104.54184	84.507777	49.066682	50.751809	79.304373	—	16.297316	0.50533	0.092028	0.086307
181	1099.988816	1339.541999	—	1298.959203	283.552827	50620.243775	1343.42771	—	37.726474	1.45529	1.44948	1.30921
182	17.420909	18.223463	17.726554	17.610638	17.738725	18.137089	19.04473	3.555107	40.488224	8.57744	0.685537	0.635251
183	18.831564	19.533312	19.226732	18.411536	19.481751	19.077939	20.563975	1.912364	15.282442	0.043248	0.041442	0.042017
184	16.693324	17.924859	18.009435	17.046748	17.760019	18.162627	19.838211	1.329779	10.485753	0.045017	0.042191	0.042057
185	16.838626	18.30742	16.687492	16.48327	17.76081	17.966784	19.283392	1.056395	9.570745	0.037852	0.036121	0.034796
186	16.371669	17.262753	15.996938	17.003435	16.603681	17.277025	18.749722	1.024019	31.635335	11.7023	44.4207	43.3477
187	18.162474	19.247236	17.226007	17.618565	18.978193	18.617761	20.291237	1.262775	14.517983	0.043319	0.042594	0.040045
188	17.299222	18.078718	16.694931	17.142624	17.78969	17.929119	20.13849	1.060398	12.71177	0.045191	0.04575	0.043304
189	17.098631	17.764284	16.863465	17.282719	17.833119	18.188274	19.624317	1.447021	10.943134	0.037247	0.037597	0.03595
190	102.844583	107.376728	414.799815	118.721518	120.616705	468.485152	109.217321	1440.151349	8.720145	—	0.019693	0.018689
191	19.724249	19.838913	20.058389	20.0024	19.863295	20.464915	21.605493	1.98714	17.898014	0.041986	0.043475	0.037677
192	64.484267	68.220799	1443.427617	75.848896	45.623076	434.176451	69.476025	—	18.138358	0.293911	0.096765	0.087047
193	18.445927	18.858582	18.311312	18.687885	18.644803	18.59349	20.553768	1.159279	12.275127	0.04505	0.041793	0.041783
194	140.927293	151.180504	8148.473772	155.612571	69.646114	2004.405937	154.311214	—	21.749533	0.895042	0.244078	0.180151
195	390.055238	418.518051	60698.175006	419.564038	131.394795	14563.523603	421.942604	—	24.951592	10.0451	0.421613	0.408385
196	19.492123	19.044938	19.140407	18.81328	19.052504	19.167014	20.646346	1.707934	12.919456	0.042345	0.041406	0.048326
197	18.414884	17.43707	17.63608	17.511803	17.810723	18.696841	20.215207	1.503153	12.797254	0.045701	0.04186	0.043582
198	16.746141	16.559429	16.252501	16.506082	17.868703	17.284427	19.03703	0.671854	41.107514	8.81092	0.699691	0.67508
199	18.226898	17.821209	17.84567	18.906024	19.126668	18.456682	20.579462	1.785686	16.670018	0.040645	0.041672	0.038189
200	17.356273	16.895382	16.933841	17.920577	18.614833	17.561487	20.043188	0.999272	12.787913	0.042083	0.044345	0.03922

Table 18: Results on *Random Syft 02, 0-100*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	1504.973108	5157.85633	—	1857.615553	2446.271323	6742.374297	2886.465606	—	4590.170819	21.8974	978.013	7.64967
2	18.706711	18.533534	19.349636	18.803158	19.292196	19.238666	20.782447	1.245319	14.055715	0.022093	0.023939	0.021903
3	3552.730691	290.086476	—	999.373529	276.591712	15301.785696	291.288737	66763.036325	14.90648	0.076225	0.078319	0.062949
4	4775.026745	977.8761	—	3626.136501	249.937242	250056.665951	987.554438	—	163.610671	93.8458	13.697	11.1197
5	2450.454876	2829.232249	—	2654.521757	1876.844986	—	2835.214667	—	321.614877	—	397.263	347.974
6	9297.348399	1830.678317	—	2201.321798	1737.504453	24281.761385	1844.30975	—	20.38892	0.039756	0.034423	0.034081
7	18.558617	19.291644	19.592553	19.768541	19.282277	19.088439	21.151468	1.419	12.99837	0.035211	0.032893	0.028164
8	647.184126	724.587044	—	403.818041	251.998173	55634.419844	731.882461	—	170.913881	22.4828	72.3822	66.4217
9	86.591293	206.334595	9057.499966	95.894753	166.545235	5576.509296	153.467425	—	120.325054	21.2469	13.884	13.8187
10	10681.127172	11508.697939	—	11586.290259	2169.267205	—	11616.413402	—	24.931001	—	0.019879	0.023657
11	6232.394258	793.684201	—	1334.331733	889.277276	6970.327383	802.275172	—	20.162265	0.029979	0.029274	0.027779
12	120.501293	39.509624	3030.497785	105.571956	37.771401	600.520208	41.770665	178109.056765	19.04953	0.294271	0.219959	0.163034
13	16438.349097	2877.949597	—	3996.236484	2524.975001	275583.60457	2885.456476	—	44.592512	0.566023	0.107322	0.091596
14	21.72036	28.729516	63.564985	22.849808	28.273522	59.178203	31.236353	6878.638779	9.829246	0.158144	0.074229	0.084331
15	4074.917305	17218.998067	—	1166.417746	991.690724	4385.690539	7665.608828	—	17.052352	0.042633	0.032727	0.029416
16	10535.661628	11344.875092	—	11561.475574	2079.996579	—	11501.856079	—	16.129658	—	0.019693	0.019956
17	26.551993	100.839968	602.193323	27.823097	100.819801	511.44433	45.638522	418.903229	63.302181	36.0567	293.692	258.126
18	404.126927	141.9924	2239.744404	195.287627	73.121931	1571.616317	144.290289	967.398508	7.798195	—	0.020437	0.024088
19	10389.196961	11229.82228	—	11477.037008	2129.519923	—	11361.660461	—	445.725844	—	139.492	128.213
20	8642.807414	9167.815582	—	9522.203496	2008.533157	—	8741.1303	—	14.929145	—	0.021445	0.020482
21	187.094327	199.639336	7264.601276	131.095429	89.311697	2416.974799	200.613347	—	24.694945	0.886062	0.177933	0.146722
22	18.924151	19.000528	18.574103	18.921174	19.257784	18.815346	20.424527	1.337612	339.39851	3.13836	0.34577	0.324545
23	18.98905	18.400137	19.293014	18.276016	18.891643	19.313683	20.953118	1.37052	14.08606	0.055303	0.03327	0.035561
24	108.354634	113.142386	1419.414927	123.127695	75.171342	1034.848817	116.133575	4458.441239	8.434413	—	0.019833	0.018403
25	2432.074252	4089.532827	—	2701.373821	2020.822758	—	4149.733783	—	277.728663	—	338.434	303.588
26	1519.416103	1584.411542	61863.592531	893.667088	272.29408	14673.139792	1600.753265	—	10.462261	—	0.02477	0.062023
27	17.230536	17.124773	17.458637	17.00102	16.876577	16.818185	18.823343	0.420772	9.359446	0.076324	0.130935	0.120969
28	1024.592862	1101.013751	136546.488286	665.1213	347.874298	30140.703774	1106.010372	—	28.375347	10.2657	0.658548	0.575248
29	5249.935978	5760.815287	—	5829.483975	1053.684258	—	5785.626175	—	4486.779718	—	42.5111	74.2423
30	2971.49795	1824.321655	—	2254.604179	491.149902	79669.837418	1839.001199	—	132.323575	5.48546	28.8763	26.7994
31	9304.898339	9875.69759	—	10253.752576	3989.189179	—	10056.280016	—	15.186167	—	0.020551	0.020091
32	19.248901	19.303022	19.216811	19.028952	19.456821	19.919151	21.134376	1.463963	15.381085	0.05454	0.041853	0.040218
33	18.841562	19.642667	18.683661	19.923757	19.157135	20.178438	21.505402	1.5154	17.769491	0.034067	0.031674	0.030004
34	159.455997	61.016805	542.977231	94.955849	41.482128	488.256295	64.536128	16525.022981	6.452843	—	0.020778	0.021335
35	18.513988	17.55893	18.766487	18.681309	18.482083	19.148755	20.465321	1.055855	25.52176	1.93097	4.39936	4.13041
36	1322.007939	511.235385	—	521.862967	274.964506	24815.806648	515.264016	—	19.9039	0.598744	0.12225	0.102774
37	19444.556071	33208.941125	—	4285.478699	3000.032166	55877.869266	33894.047594	—	20.10736	0.049926	0.035709	0.037859
38	1166.596019	1253.555684	—	1281.193365	284.96169	34162.225074	1262.033541	—	106.942367	8.16926	2.55561	2.46503
39	20.474829	20.606662	19.532546	20.017287	20.206729	20.365065	21.709426	1.640538	14.6635	0.034797	0.02434	0.023653
40	20.383671	20.325921	20.084979	19.494384	19.014032	19.658419	21.516486	1.594364	20.013454	0.056009	0.030512	0.031589
41	21.409509	24.482057	111.855449	21.432505	25.606779	82.033509	27.280313	225.306216	46.981821	1.2478	9.43219	9.35792
42	6969.947326	1322.411911	—	1493.134562	1436.008585	4998.30372	1340.290285	—	18.897928	0.037135	0.04114	0.035312
43	2784.55414	621.307346	—	819.098998	505.064143	31306.872391	631.90935	—	19.174882	0.190449	0.063286	0.062209
44	904.714361	1005.214738	—	1032.801646	233.994698	71645.452769	1003.578073	—	317.381621	95.5719	2.60936	2.74642
45	6234.460796	6847.201473	—	6991.655589	1242.04586	—	6958.9953	—	4810.02683	311.977	46.5761	44.2941
46	1049.039503	395.772375	137478.244692	386.776732	255.787234	4275.473317	402.326423	—	13.257913	—	0.022126	0.021029
47	117.913823	135.912274	6963.115274	131.075676	99.223301	3838.904704	140.051575	—	18.230485	3.67986	1.13069	0.978477
48	7193.98618	7584.857998	—	8022.879379	8268.005696	—	7812.158916	—	12.76014	—	0.02024	0.018588
49	21709.370539	110653.482037	—	19189.453175	4007.617853	—	41944.736449	—	10000.403112	66.4859	—	58.6544
50	48.295809	27.669747	562.411484	39.290156	28.51009	53.068544	30.386979	394.740534	10.080901	0.028552	0.060121	0.024461
51	16667.285203	4340.890991	—	4837.970497	2587.736722	—	4441.300195	—	29.838471	10.656	0.43725	0.401334
52	3955.6497	958.850776	—	1009.192279	976.760532	3227.148247	978.527533	—	15.63058	0.042461	0.03357	0.030235
53	18.61114	18.879499	19.033662	18.953912	19.155658	19.028724	21.197601	—	10.342269	0.139157	0.07362	0.071585
54	17.494366	16.01538	16.71926	17.64128	16.92803	17.626395	19.689766	0.61638	10.238338	0.449211	0.352726	0.316135
55	18.20183	18.262187	18.201022	18.666407	17.727441	18.831151	20.415424	0.88687	22.452138	0.876173	0.189905	0.169749
56	18284.708516	19134.33934	—	20271.461628	20439.294565	—	19610.911741	—	15.318891	—	0.021259	0.020551
57	1204.253097	256.597478	—	352.911412	250.047634	3828.321192	260.909678	—	13.012331	0.056238	0.031704	0.032352
58	3878.942162	4492.442517	—	2191.933232	1187.205825	—	4577.414072	—	2782.164848	394.597	—	29.2764
59	13603.906262	8850.114969	—	6878.292901	1932.926927	—	9005.543826	—	1141.847714	—	184.311	38.7778
60	18691.111057	22483.012946	—	19969.556227	3337.307838	—	22447.026553	—	391.444578	95.9632	—	68.8754
61	19.185424	18.225312	18.922194	19.184921	18.867766	19.719237	20.99999	1.245065	10.523151	0.027256	0.07191	0.024729
62	17.612252	18.186858	17.535461	18.199539	17.892125	17.911882	20.337714	0.97924	1677.207654	931.331	—	60.8032
63	2009.408461	2140.679113	197397.750008	2294.759698	539.831269	43133.142434	2175.441184	—	12.678589	—	0.022118	0.020863
64	128.606121	117.431946	2553.95682	130.317207	118.927534	1990.773653	120.246206	3718.285793	167.700893	41.0649	235.953	212.784
65	2583.61898	2241.621427	297094.040307	1135.772469	727.025187	35792.928971	2267.764591	—	12.878018	—	0.021423	0.020742
66	449.32368	146.366566	107918.606157	196.85044	128.033587	2351.508069	150.281915	—	11.531659	0.060111	0.031927	0.031314
67	1209.291928	475.085841	—	461.911902	292.960166	11885.811729	482.884799	—	17.744852	0.11685	0.065564	0.048762
68	33.380586	29.908496	93.044121	31.202259	28.580185	83.582502	32.179876	172.232818	22.365919	0.301464	1.01638	0.980094
69	20.967313	20.772257	20.331724	19.727171	19.956186	20.469147	22.340343	1.669579	18.086534	0.054305	0.049434	0.038313
70	367.492826	171.206615	1642.108847	391.287344	170.549052	1297.513347	173.918609	33458.901054	117.252065	489.027	3481.1	3257.81
71	24.99341	126.121817	903.129466	26.070249	127.063131	599.086324	41.825819	8607.4523	442.945008	92.4947	8756.87	8207.41
72	225.629862	229.347425	2029.810203	185.267201	177.936091	1764.299924	235.169514	3038.423848	8.960939	—	0.020733	0.024909
73	90.271649	345.549383	18935.034787	101.865673	298.060082	10525.537704	168.097165	—	396.946236	50.9325	95.1075	88.5057
74	19.635722	19.582577	18.804487	18.817287	19.013054	19.933917	21.64452	1.291582	17.694893	0.11648	0.086697	0.077285
75	777.593488	850.007288	71693.806024	797.497498	870.664316	72271.360916	865.543295	28850.313122	11.838017	—	0.020589	0.022398
76	200.468891	90.427614	26706.79078	117.358515	77.43714	1199.468193	93.83898	—	10.286667	0.065938	0.043595	0.039314
77	124.568559	79.204374	9205.566516	93.563207	63.153803	517.667698	82.90211	2302.659393	10.453178	0.082922	0.0647	0.061305
78	239.68773	83.19051	7901.846934	144.312656	66.86581	1112.989934	87.473142	21337.836906	11.012287	0.147095	0.087841	0.081264
79	18.627963	18.457442	18.352323	18.731774	18.49713	18.731527	20.697632	0.991003	11.619935	0.050706	0.038443	0.033486
80	19.049643	20.166132	19.18098	19.755055	19.693606	19.653264	21.934615	1.418644	15.003794	0.052287	0.036515	0.038543
81	356.635406	375.867005	13528.037915	374.745769	149.993626	3193.757443	382.055253	—	9.699199	—	0.022196	0.025043
82	23746.746214	6399.217954	—	5135.413935	1430.660711	144253.553327	6485.921791	—	18.442181	0.080087	0.080569	0.065623
83	19.429648	19.535482	19.370947	19.1612	19.221848	19.181263	21.42699	1.259022	13.012719	0.023214	0.022405	0.024159
84	15931.778242	5907.625065	—	5265.857592	2547.234509	—	6002.649927	—	358.081501	25.2855	1.25112	1.03136
85	3550.203409	3862.667402	—	4110.276795	968.848053	237490.484923	3863.361022	—	12.71882	—	0.020995	0.019676
86	17.426406	16.925029	16.810167	17.292758	17.669078	17.290109	19.602863	0.477387	11.102065	0.082534	0.273026	0.166964
87	34.715151	28.307099	72.006696	34.322875	28.83422	94.973141	30.633266	429.933907	33.173695	3.05685	2.20391	2.18218
88	220.159545	222.199615	5433.40993	222.003631	221.235334	5422.083831	225.869362	1816.188841	10.842513	—	0.022756	0.018413
89	17.837425	18.053809	18.048761	17.699905	17.800165	18.618625	19.702445	0.747664	8.619019	0.049015	0.038805	0.036929
90	108.382286	146.428219	1559.755128	62.094148	70.229878	431.975927	151.219893	23802.570888	14.852793	0.438277	0.133108	0.123532
91	11863.633416	670.161033	—	1071.22681	372.299212	5891.816224	677.600063	—	15.115988	0.070214	0.050211	0.044162
92	23629.802256	7321.933132	—	7063.506575	7028.318803	39253.955018	7401.736602	—	24.578779	0.081757	0.053515	0.047798
93	70.43492	73.848192	1106.648256	72.390541	71.77538	500.713685	75.743672	5808.641036	24.768374	9.90021	0.990539	0.962736
94	137.392352	327.15854	7024.158072	150.682286	343.767785	7146.513298	263.673748	26836.05506	9.788042	—	0.02116	0.019072
95	4335.797672	4663.97851	—	4878.208593	976.523467	—	4761.433108	—	1556.641373	—	45.2737	40.3999
96	18.693188	18.98534	18.901924	19.274475	18.956925	19.176934	21.186712	1.545231	12.1075	0.029937	0.026583	0.024163
97	59.444051	56.814887	2770.254471	62.258289	49.475093	720.415206	60.562269	58108.463046	29.122142	0.38621	1.79193	1.80976
98	19.101489	19.332099	20.014669	19.978005	19.740154	19.961741	22.258343	1.888736	16.533036	0.085834	0.035732	0.031881
99	17.962762	16.856812	17.149568	18.312306	17.994335	17.928682	20.359847	1.207551	9.143653	0.027502	0.025445	0.034946
100	9146.47612	—	—	7072.342539	5795.497969	—	10862.566981	—	272.586767	63.7672	70.1691	12.3616

Table 19: Results on *Random Syft 02, 100-200*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
101	1893.295514	2124.460514	—	1938.579287	1702.496663	—	2154.451071	—	434.622776	—	655.528	107.261
102	2319.958785	846.097249	—	844.271099	518.399692	22340.772101	858.199308	145879.742948	12.540078	—	0.022875	0.024112
103	144.770494	165.784182	7679.685127	158.743126	116.606741	361.039836	170.732243	—	97.028037	1.5045	2.49803	2.38788
104	18.659551	18.373764	18.109879	18.437595	18.390078	18.653903	20.269547	1.430386	10.908449	0.083266	0.045452	0.065573
105	133.218373	165.405762	12514.192894	97.025729	69.989218	605.606859	174.44256	—	10.511282	0.078108	0.05796	0.056922
106	62.570635	59.888824	790.981212	62.905215	61.995424	586.848967	63.477016	1159.876483	161.391046	30.2802	10.8114	10.4937
107	603.904043	641.253886	110485.987951	621.13678	621.855499	73897.325208	653.651782	—	330.434515	352.153	261.918	243.038
108	18581.952243	19053.028961	—	3894.69085	3144.082299	45178.40042	19617.251982	—	32.155439	0.051518	0.047105	0.02895
109	322.742314	107.464257	7343.486179	210.92186	66.714149	2192.387368	111.058221	—	20.780582	0.892174	2.55266	2.3162
110	2793.674467	2956.476709	—	2824.535447	2911.305814	—	2977.935352	—	27283.088914	—	—	200.262
111	17.556391	17.290686	17.371601	17.469615	17.183872	17.307855	19.304667	1.135934	11.546653	0.259431	0.156717	0.139262
112	1660.138427	2055.670734	—	1048.220752	569.571937	256801.197538	2088.175957	—	582.059454	1203.95	161.263	152.637
113	1163.112365	1245.85179	—	1285.179322	285.133541	34237.344325	1262.631791	—	107.896886	10.7745	2.54548	2.69736
114	750.029753	2289.063798	—	581.518801	365.847412	5402.651219	1392.136485	—	14.414567	0.175487	0.141569	0.119784
115	10693.369505	11509.979758	—	11783.815718	2188.126635	—	11760.873203	—	3115.907412	—	120.647	113.506
116	506.128836	541.053207	167453.375152	596.657327	150.629919	8429.15418	547.011365	—	51.472661	2.58593	0.801641	0.766391
117	19.061023	18.958967	18.761046	18.958723	19.101531	19.219031	21.083585	1.765207	23.796874	0.302163	0.117473	0.088349
118	93.104075	89.11147	2122.936073	83.873066	78.478125	1155.189164	91.956726	19237.218082	9.286089	—	0.020131	0.018569
119	440.807987	1146.292145	247211.907434	473.511263	782.72827	93787.413506	850.329553	—	185.092912	428.215	316.074	282.526
120	324.450883	358.766301	24801.826211	344.09409	119.91857	9926.409959	364.107634	—	10.601809	—	0.021203	0.021725
121	1557.17222	8690.244926	—	1439.946105	671.588515	91520.197268	2917.338481	—	307.63509	6.04039	230.78	190.964
122	18.827312	18.810114	19.200876	19.299457	19.828795	19.647516	21.707486	2.047421	13.474204	0.024283	0.025784	0.027236
123	13418.081894	14313.49923	—	14166.254903	2469.992358	—	14573.282937	—	16.225014	—	0.022492	0.021
124	238.518254	620.763738	76932.764954	253.04159	253.365045	5072.954847	453.689307	—	61.503684	1.1293	5.35618	5.09353
125	36671.70772	1081.049377	—	2467.468681	273.853646	2180.573976	1095.083289	—	12.752069	0.045941	0.025633	0.02692
126	2520.031135	690.181101	169350.219546	1078.599959	203.2541	39768.687156	705.183819	—	62.970359	13.9433	4.14191	3.94596
127	56.579959	68.609626	1592.897634	68.230954	51.139625	287.31449	72.27534	—	10.616078	0.117838	0.199226	0.19051
128	453.144634	1008.103438	—	220.731603	200.590131	3566.068463	849.761536	—	14.049378	0.079409	0.055745	0.051642
129	20.297997	19.860686	19.95314	20.391722	19.822262	20.797076	22.357116	1.858983	17.976851	0.026479	0.025392	0.02748
130	256.486378	271.106166	7650.730453	272.87958	118.010589	4985.0397	277.945204	—	9.351083	—	0.019071	0.019738
131	17.331404	17.947864	18.013265	18.04269	17.672756	18.326432	20.013773	0.729705	90.912479	7.00272	1129.04	1034.04
132	22.203293	52.546093	488.437332	23.135442	53.199206	383.165014	37.240201	1103.248408	111.502757	119.236	1347.79	1230.21
133	16.269576	16.639573	18.414655	17.505682	17.197829	17.987799	20.19382	0.617601	67.064782	3.69534	13.4343	13.2136
134	1119.069705	1247.872149	43739.047134	772.38457	936.716453	30018.712714	1273.590533	—	12.844641	—	0.020405	0.019785
135	19.964301	19.735497	19.872711	19.948527	19.955943	19.882377	22.599307	1.932033	19.600965	0.072705	0.043363	0.040022
136	17.400066	17.030299	17.538979	17.573989	16.636702	17.462557	20.244655	0.721784	7.962425	0.046583	0.032631	0.031587
137	17.799578	18.420587	18.425099	19.299427	18.424741	18.509158	21.970846	1.340391	13.545084	0.056421	0.033062	0.031675
138	18.575722	19.066193	20.447542	20.013841	19.845466	19.672203	22.66037	1.855618	22.133657	0.024905	0.025523	0.023365
139	87.12556	77.929919	2439.087484	71.782341	53.294797	348.220088	83.522583	4539.820639	7.857572	—	0.026101	0.019873
140	18.961644	19.70638	19.433043	19.157042	18.773367	19.389279	21.875079	1.369682	11.947425	0.023506	0.02474	0.022579
141	18.968232	18.444068	19.496264	19.3366	18.567722	19.06394	21.630268	2.068988	14.70138	0.057739	0.027245	0.026235
142	1562.276252	8663.909709	—	1433.28004	671.100111	91408.481569	2935.517799	—	306.6649	5.9713	212.462	201.513
143	17.482322	16.894645	17.950643	17.572241	17.678651	17.94416	19.45001	0.629095	12.643728	0.774173	0.602218	0.548826
144	3871.608361	4073.848668	255529.746128	4367.329742	1623.740803	215816.2112	4143.724794	—	13.393991	—	0.023676	0.019465
145	61508.307891	8000.421511	—	8091.42799	7382.412823	48618.943092	8090.019175	—	25.067239	0.08741	0.067437	0.063397
146	2003.651185	1269.266757	44344.314239	1016.759551	924.47249	19068.79487	1286.483198	—	12.708032	—	0.020151	0.019289
147	87.305802	65.581928	395.954988	68.381499	51.224319	436.571172	68.109885	209.369222	8.38179	—	0.019722	0.022764
148	18.037622	17.597687	18.569963	18.22432	17.698549	18.206054	19.893386	1.134664	16.526436	0.274908	0.2003	0.158469
149	19.355999	18.033032	18.41511	19.291566	18.756131	19.960798	21.148663	1.251684	12.877941	0.029778	0.026794	0.027145
150	84.448715	151.407813	6765.250901	72.912806	86.831925	393.757766	153.496304	—	10.503384	0.063149	0.047458	0.045684
151	3744.658726	20333.524252	—	3442.480905	1097.100708	189495.823578	7110.047102	—	737.181534	12.5258	475.563	435.339
152	18.214688	16.585916	18.326859	18.037068	17.972869	17.904444	20.340294	0.935692	487.791834	82.6521	66.8504	62.373
153	18.247536	19.12657	20.083726	19.565331	19.288319	19.279591	22.033812	1.54521	15.669933	0.024932	0.027162	0.022971
154	18.498561	18.096996	18.721186	19.328503	19.770436	19.670747	22.23537	1.264265	27.532433	0.286333	0.107379	0.091185
155	7252.810102	8798.484896	—	7844.155042	1428.113067	—	9011.163913	—	906.072538	14.2613	206.812	187.362
156	572.087686	173.114973	167899.776234	215.627096	151.007894	1709.729615	176.949038	286015.364551	13.384829	0.054649	0.033825	0.027348
157	83.248256	30.76489	2237.874494	47.885102	32.802423	141.470102	33.680896	521.916687	9.352752	0.037447	0.03062	0.031478
158	6248.167087	799.736758	—	1325.532016	887.849494	6931.900882	806.873732	—	19.320009	0.035002	0.030806	0.028527
159	16.54009	17.795498	18.033105	17.884388	17.685597	17.735938	20.174523	0.849426	10.709011	0.10109	0.075234	0.067039
160	218.054669	225.395863	2668.750154	234.490695	240.020209	2750.609021	229.060529	5984.972326	8.560967	—	0.024088	0.019312
161	1887.905482	2035.753627	—	2128.192337	456.154257	195742.146204	2069.21078	—	13.140964	—	0.019233	0.023745
162	24.474929	69.743711	765.730148	25.664107	70.889679	375.92149	41.153659	9538.621593	274.186257	74.8452	1497.73	1390.83
163	18.205906	19.013974	19.789232	19.788377	19.402011	19.433696	21.504058	259603.395033	15.882312	0.124762	0.062841	0.059927
164	12985.777093	6792.482676	—	4022.689273	6775.120583	23654.671747	11111.188488	—	20.814716	0.057976	0.043975	0.037036
165	18.106679	17.860624	18.069532	18.250189	18.249112	18.228034	20.05165	1.046052	9.807512	0.025262	0.032629	0.03871
166	1534.249197	1785.505537	—	1001.252131	530.842378	185399.937362	1820.219424	—	354.835038	69.24	157.558	6.3675
167	16955.528934	2960.858761	—	3986.209745	2784.731247	119864.028568	3012.407971	—	30.405663	0.197832	0.060413	0.053287
168	61.263992	63.062745	670.791093	63.904099	64.84201	676.518251	64.897966	415.492053	8.327135	—	0.019657	0.020819
169	769.874747	3663.990336	—	590.316632	385.80413	11133.915241	1438.971334	—	14.377549	0.170903	0.14449	0.131732
170	2447.596788	2416.167282	—	2862.114346	543.992728	—	2460.62894	—	444.626651	398.9	37.3955	34.4995
171	20.318521	19.825925	20.041818	20.470964	19.706759	19.895546	21.944788	1.598868	15.590935	0.026156	0.025404	0.024914
172	6861.531362	1240.031394	—	1684.859559	1235.016231	11432.234105	1249.445473	—	19.002806	0.032032	0.028518	0.025343
173	19.431865	18.964024	18.872808	19.249141	18.954118	19.393496	21.095446	1.207846	11.060395	0.05889	0.037824	0.034852
174	8399.467604	136368.306472	—	1961.254819	3434.8358	26361.705929	16036.95932	—	14.863648	0.023035	0.022007	0.028874
175	20.39012	20.01375	20.179135	20.389824	20.442884	20.357387	22.256898	2.010696	19.306095	0.051426	0.030485	0.029382
176	444.448675	114.269532	43006.827247	327.36741	57.510683	2848.599313	118.04581	—	41.036203	0.268135	0.445849	0.473477
177	24644.011233	23109.386453	—	14749.645029	5733.108809	—	23663.261367	—	494.981389	2.13438	1.91645	1.86911
178	1083.085537	247.223854	236945.023807	328.248064	252.994388	1991.111028	252.978785	39455.456817	12.78484	0.018993	0.021347	0.021523
179	4087.141365	4402.065801	—	4570.853776	937.371562	—	4482.277197	—	15.708868	—	0.020381	0.019005
180	17.717758	17.129116	17.801177	18.286782	17.740305	18.121657	20.204427	0.942595	9.063687	0.057409	0.042091	0.049349
181	94.312492	34.419263	3067.786161	64.632634	44.526318	80.147739	38.403303	7300.489236	10.361032	0.038387	0.032492	0.033308
182	58.909461	63.53549	1459.339026	70.30898	58.693066	764.847473	67.321623	—	21.335163	0.937392	0.254591	0.22619
183	5438.985266	6933.124619	—	5869.082228	1584.711857	—	7044.187287	—	1084.008903	89.3043	—	72.804
184	17.974956	17.604699	18.225361	18.205567	18.102789	18.138763	20.220323	0.963797	142.248974	98.6547	119.5	117.382
185	2599.872649	2239.054616	298560.364325	1130.590665	725.099306	35827.873954	2269.801569	—	12.26315	—	0.021574	0.024855
186	18.235893	18.956267	18.582173	18.908974	19.113273	19.231068	20.859973	1.192543	16.301141	0.109756	0.096187	0.070894
187	8034.055522	8413.509611	—	9016.949259	9104.148836	—	8528.469025	—	14.185462	—	0.021335	0.018832
188	15720.015718	17017.987534	—	16562.387396	2777.074031	—	17330.890842	—	430.484482	219.924	—	22.4331
189	7010.763393	4993.441765	—	6423.034665	950.719154	—	5088.930808	—	106.509693	8.20707	53.9162	49.5487
190	104.354625	61.522434	2522.214944	82.739685	51.480849	1244.5166	65.2019	1146.297857	81.225863	0.591436	15.7432	15.1563
191	19.43745	19.907392	33.883081	20.556387	20.051552	28.364187	22.116688	38.911868	14.276463	0.094043	0.520541	0.46681
192	25.850773	245.703502	2318.133847	36.623975	247.091989	543.416613	46.35037	—	191.854669	58.7659	1561.92	1416.25
193	546.188391	221.932079	105716.732423	248.376202	143.962472	7360.092713	226.720461	—	17.369051	0.23062	0.067343	0.066995
194	126.383501	128.379448	216.356056	94.48065	86.522107	232.209295	131.78043	1292.598908	8.42389	—	0.020408	0.019688
195	23038.51963	—	—	4953.251996	5683.807077	65565.887905	43784.778239	—	21.28359	0.037978	0.049457	0.03369
196	16972.804761	17795.923146	—	18572.343079	19182.642253	—	18079.836417	—	13.89295	—	0.021158	0.023478
197	17.914756	18.417526	18.413838	18.285834	18.071416	18.397896	20.025909	1.008716	8.511284	0.027657	0.023506	0.028131
198	5436.900779	5841.854009	—	5832.70619	1122.046241	229762.731348	5874.823302	—	14.345949	—	0.020461	0.02347
199	46.73356	51.007011	362.852413	49.985269	52.384179	369.307416	51.902771	270.988981	7.759925	—	0.019915	0.018336
200	18294.477602	10913.063188	—	5894.223411	3385.880089	—	17562.290062	—	43.436186	23.8899	0.990356	0.85296

Table 20: Results on *Random Syft 03, 0-100*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	5188.103402	132263.301164	—	4180.54409	2711.475735	—	5342.515073	—	78.270212	1.10837	6.9235	4.49524
2	105948.080231	1316.554382	—	25574.821837	1300.172417	—	1346.357127	—	25.187672	0.220906	0.450101	0.396162
3	19.265656	19.662589	20.12297	19.936581	19.238229	19.713098	21.89371	1.448557	14.246965	0.063966	0.029014	0.033395
4	233.067626	243.093066	42490.74027	171.539432	186.165276	1894.872473	254.627278	—	13.381828	0.159445	0.162832	0.163528
5	45.987631	43.572294	43.56121	49.200354	47.383417	47.982572	47.897064	351.33871	19.11144	3.26319	5.61274	5.38125
6	16225.575144	1317.145448	—	3527.678918	1404.144196	48471.777664	1347.228569	—	26.177785	0.066827	0.043084	0.039006
7	491.301194	227.206716	157146.942522	204.606569	88.506694	2392.068684	234.019052	—	12.294267	0.061251	0.043303	0.031415
8	7187.974353	7621.506997	—	7726.176755	1308.297586	—	7819.265315	—	21002.995631	384.041	—	48.5037
9	3903.415906	902.45203	246099.290463	3714.069421	935.838864	205175.529418	925.958384	—	12.551581	—	0.106661	0.019967
10	1926.266914	544.98582	—	1058.920526	213.272974	60194.582035	558.747681	—	703.588207	100.435	513.279	7.60559
11	32.393772	133.87675	280.365328	40.486832	136.123275	397.773856	58.533279	7877.729223	68.959669	19.7827	134.174	116.071
12	114.601265	70.563451	3965.29635	86.676466	61.789695	286.836891	75.978006	—	9.056456	0.04561	0.032771	0.032238
13	16.742854	16.833951	17.318593	17.446383	17.290594	17.136917	19.634112	1.048952	26.806175	0.415625	2.92472	2.8255
14	447.88612	125.286317	36818.560445	227.610468	92.651685	2963.69477	130.201464	—	11.229178	—	0.021606	0.021899
15	81.702213	81.709404	3025.26317	94.550918	93.438311	3234.99572	86.08585	6896.689925	8.971957	—	0.021226	0.020501
16	64.675298	566.998004	40483.808778	68.587705	566.764283	6033.465967	119.185958	—	155.860764	11.3891	514.987	472.484
17	24253.162615	17633.511367	—	10457.292991	4109.395691	—	18424.064685	—	39605.754749	—	—	160.528
18	27398.615095	17681.888495	—	19440.468703	3136.588642	—	18420.072258	—	11285.037872	—	—	385.615
19	705.310967	295.721426	156372.035116	327.783939	193.463556	8838.871509	305.038122	201334.030296	24.243047	0.176193	0.389631	0.343872
20	866.869609	611.527261	613.204937	795.321798	687.746504	690.685853	684.251288	—	14.652677	0.099089	0.108866	0.095907
21	24262.949926	8077.107961	—	6088.529978	4524.733224	152505.712774	8258.635133	—	16.404068	0.064872	0.046853	0.040966
22	2875.969403	494.564016	—	979.418292	362.99482	35349.615704	504.60767	—	17.919814	0.193951	0.119175	0.106673
23	10741.172878	1552.87021	—	7590.420448	1486.210677	261019.946385	1597.041116	—	52.981284	5.96844	16.823	16.8125
24	13908.973493	13857.768371	—	3265.783669	2490.041992	110726.813524	14270.917986	—	19.209188	0.104807	0.037953	0.035551
25	5612.124865	5305.731519	5271.322945	4719.55815	4545.995774	4563.167434	5923.171722	—	33.272102	0.533558	2.07766	1.97717
26	39.774385	215.008261	913.700459	38.919523	212.273371	1063.564356	71.141653	3142.028738	7.895257	—	0.021943	0.023455
27	17.795033	19.068347	19.2772	18.484254	19.329324	18.94861	21.425516	1.28562	13.199203	0.024995	0.028315	0.021239
28	9100.124189	9704.634548	—	9950.295623	2285.476192	—	9941.550457	—	13.397989	—	0.020976	0.031773
29	2760.819133	2297.352538	—	1429.892457	730.632321	54783.663963	2317.226833	—	13.521183	0.065866	0.056464	0.05303
30	3060.53097	2555.164525	—	2772.268379	2154.349232	73832.980579	2569.791781	—	166.53304	4.29509	8.67329	8.69821
31	2020.181293	2352.85466	135202.107443	786.750043	326.430818	20733.380992	2387.500471	—	10.000203	—	0.022614	0.021994
32	19.803336	19.197516	20.183863	20.468526	19.831914	20.63374	22.093122	1.786628	18.549058	0.059556	0.031073	0.029407
33	4652.762127	2603.244519	—	2139.477858	810.064417	239104.291019	2633.56519	—	32.436064	13.3393	1.07155	1.02289
34	6567.251089	6873.185586	—	7303.03139	7529.98848	—	12791.006652	—	11.010109	—	0.02082	0.019449
35	4119.774704	397.144603	—	1553.697113	232.885011	45887.214993	411.379643	—	16.465219	0.424391	0.218897	0.220646
36	19.042594	19.426719	19.867453	18.976419	18.939021	18.821977	21.95617	1.221579	12.062694	0.064642	0.048923	0.040546
37	3218.148149	1218.677647	—	1180.694413	660.817335	37267.271377	1256.084589	—	13.236037	0.212089	0.045813	0.048694
38	4879.705248	3490.119883	—	3567.224364	825.315336	—	3556.242988	—	206.140868	—	31.4975	27.4378
39	36303.328962	1091.72555	—	9623.64872	1069.543032	70164.18722	1104.826282	—	19.621199	0.089789	0.104019	0.087405
40	4562.60434	241.539438	—	1801.074614	236.30825	16042.723269	247.878623	—	14.469067	0.029166	0.025176	0.029019
41	221773.718104	7078.61951	—	14176.314477	3698.945173	98760.140956	7212.100181	—	42.673872	1.84769	2.77968	2.77672
42	65.379993	167.209798	1746.975225	62.059114	165.235624	872.805805	121.011161	12096.081652	1321.364731	916.47	3703.43	46.9985
43	61.527722	54.972558	2502.287694	56.048218	51.495449	1140.92919	60.411041	44715.099255	272.426668	13.3003	454.471	411.131
44	27165.733126	29062.051653	—	13683.054313	1878.534726	—	29871.317978	—	20.500766	—	0.019967	0.023826
45	19.323011	18.654872	19.546097	19.608846	19.940917	19.52465	21.714395	1.53815	13.688054	0.024977	0.024285	0.02218
46	402.990242	454.808077	56934.84436	255.705959	224.441692	2464.46435	466.787074	—	13.114165	0.128325	0.149283	0.147463
47	6780.714196	372.255027	—	2001.630947	286.217683	25498.996278	379.575062	—	19.753121	0.121009	0.086988	0.075859
48	135.101681	1613.400395	30795.964209	145.276406	1623.841671	30680.33695	255.584035	96851.282852	11.295299	—	0.020674	0.023578
49	1010.01531	1087.331265	—	1120.87349	278.545162	35226.800143	1105.788643	—	47.237391	28.7316	24.2151	22.2475
50	9732.122356	10061.941908	—	9593.962119	7575.22732	—	10267.276829	—	173.201179	1086.84	381.213	347.662
51	637.908698	355.712314	—	354.041482	152.812443	13570.890697	364.546338	—	13.351662	0.945877	0.189279	0.133011
52	24.440258	96.73444	215.016986	26.32711	104.313116	311.267674	41.935894	9436.841663	51.309029	39.0554	86.1684	76.6603
53	198.500023	543.029971	11433.113397	134.198393	506.69816	6871.229211	378.280379	—	10.797168	—	0.020078	0.020173
54	1250.476569	606.422317	110769.319036	635.366011	480.214534	8338.786801	624.047366	67583.400448	14.561647	0.207999	0.167659	0.168371
55	18.049369	18.688721	18.77427	18.874233	18.576223	18.356268	21.124026	0.980078	12.778696	0.078373	0.067998	0.057846
56	84.763583	535.363913	22048.299234	96.921594	499.972761	9976.324943	157.913056	—	9.286211	—	0.019421	0.01884
57	47888.024204	81430.305778	—	11184.993533	3379.033103	84800.256956	84147.607543	—	16.969797	0.037823	0.027888	0.024028
58	3881.882266	1938.577524	—	4435.591229	436.737585	—	1962.721492	—	14.021068	—	0.021354	0.019012
59	343.231263	64.045197	33586.574231	226.162763	52.662726	3252.43086	67.015235	—	142.127575	1.29371	23.2817	20.7253
60	94.758951	200.561972	2056.784792	72.991676	193.159744	1807.325752	179.4764	21426.041358	371.509626	35.7963	4491.22	66.3832
61	315.45533	200.185702	37376.402384	185.659254	134.465823	3847.411924	205.645399	—	26.94032	0.195494	0.967774	0.84598
62	10838.82929	5923.060286	—	8065.893551	2875.983905	—	6096.108593	—	16.111869	—	0.026684	0.019309
63	7437.622306	8931.001651	—	4143.007686	2924.589396	—	9183.824954	—	16.149021	—	0.026029	0.019494
64	17.69416	17.747718	18.037804	17.554874	18.146544	17.930544	20.171984	0.748267	8.241555	0.033825	0.034825	0.032
65	5221.899229	2296.56443	—	2357.674355	1300.725739	—	2353.89615	—	175.713085	2.77468	11.1913	11.114
66	17.467616	17.497496	17.59718	17.789206	17.432247	17.708103	19.669398	0.767388	200.458165	20.946	1331.99	1247.94
67	17.230968	16.981399	18.229106	18.77413	17.751682	18.362813	21.313296	1.175862	94.081073	14.9712	6.27967	6.18562
68	15.748373	16.39222	18.045609	18.224088	16.975071	17.081986	19.884748	0.710574	7282.464025	1810.88	—	813.171
69	15.852497	17.099387	16.438122	18.087967	16.787283	16.935605	19.749176	1584.58411	66.445364	117.613	1128.4	1012.46
70	1986.320347	2113.515868	94045.844991	1907.165409	1993.169058	43813.452633	2168.562643	—	13524.615386	—	—	313.815
71	2775.754809	3051.914163	—	1542.566243	662.097241	274024.692977	3113.052237	—	201.422725	199.602	7.34033	6.71268
72	682.356981	1834.489056	—	280.223395	1700.785809	90997.001993	1317.514092	—	53.808895	23.5293	22.3276	19.3326
73	12017.490909	11184.417809	—	3961.200307	2595.917843	—	11460.266073	—	13.028386	—	0.022536	0.02326
74	26855.21214	6841.211334	—	9669.875691	2660.699542	—	6962.592823	—	12.462586	—	0.020765	0.01996
75	48.150176	744.326413	22204.1389	42.227493	752.523359	17744.960884	87.518116	—	3398.932213	5110.98	—	—
76	1100.851875	401.085573	39706.968675	1076.035798	398.591663	45369.654329	405.811727	113652.264577	592.931196	206.699	4077.9	75.8544
77	17316.924829	11222.430886	—	6517.707014	2233.736514	—	11533.41836	—	1352.34515	—	230.144	37.3085
78	1899.706821	492.607377	—	1030.001657	346.556582	20374.45049	503.839595	—	30.685097	0.25436	0.642067	0.574638
79	19.716765	19.353361	19.279231	19.464262	19.589282	19.386486	21.595342	1.346945	35.923668	0.396668	0.199439	0.170212
80	17.724154	15.964294	17.837401	16.899656	17.834614	16.992366	19.886882	0.726234	8.551355	0.027568	0.028015	0.026941
81	79.45386	60.723339	2631.107916	81.972893	64.890423	2369.413961	66.065193	694.808335	9.315012	—	0.022854	0.021067
82	1606.982268	1365.041351	—	1106.655029	476.102552	41050.080994	1357.06199	—	811.354642	16.8427	439.128	408.152
83	43915.974405	10550.386214	—	12003.900307	4470.507564	—	11244.468648	—	2574.257109	37.0065	14.1866	13.5071
84	4354.173859	2322.799714	—	1655.704684	722.744187	94082.871443	2368.684344	—	23.924779	2.01954	1.15394	1.16925
85	1978.647246	362.457297	—	830.36373	310.203616	17596.240346	371.821398	108475.565513	12.544578	0.064957	0.040562	0.038264
86	12467.581593	2923.257919	—	4160.083648	2815.948213	107405.874382	2952.479219	—	23.047652	0.213089	0.265507	0.241168
87	28.207428	62.908588	105.86551	31.288094	60.461353	106.439698	50.599684	1572.032877	65.231726	8.01234	188.203	163.495
88	4122.279012	4798.178722	—	2945.158012	1243.657471	—	4880.537416	—	14.315506	—	0.020144	0.019897
89	19.228092	18.606449	18.847573	19.658036	19.103798	19.464135	21.601989	1.307729	12.178001	0.035302	0.027536	0.022438
90	20582.032218	13156.245401	—	7829.198582	3739.45358	—	13499.610877	—	1426.590005	—	—	32.3332
91	42.21637	195.660654	2034.876615	36.467509	187.380017	1515.646013	74.482944	3415.514159	28.488278	5.6953	7.63679	7.43229
92	6902.830781	2954.760333	—	2435.903581	617.082745	—	3026.054143	—	320.349189	7.87659	0.670694	0.653867
93	2352.061205	678.986815	—	1953.155772	575.68895	239981.392441	698.452804	—	616.280426	297.877	—	96.8386
94	53.457665	56.063513	472.712665	49.557852	51.166836	355.454469	58.374138	1254.429242	15.777084	0.466024	0.226092	0.2019
95	26.16463	24.116394	227.070036	27.778917	26.066628	61.044051	27.775256	359.673758	60.25719	1.469	5.0309	4.93839
96	527.046429	1950.064476	32022.874137	527.358873	1958.215973	8889.282213	1014.525626	—	9470.276692	—	—	820.594
97	518.340369	327.47001	10483.415883	406.639017	183.659189	7189.665346	331.926341	21812.749276	10.033403	—	0.067819	0.019266
98	9314.932917	3533.488929	—	2932.104508	1628.060468	—	3574.106978	—	232.632161	5.6222	0.380567	0.375635
99	28.899465	32.493796	198.937913	29.80466	34.06762	76.21039	35.504701	6320.540737	133.386275	0.697653	136.849	126.866
100	1518.781361	4658.010297	179855.565164	488.149134	1686.29256	82781.842496	2981.330071	—	10.819517	—	0.020573	0.019856

Table 21: Results on *Random Syft 03, 100-200*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
101	371.12851	147.818981	10690.126604	397.076257	138.739578	2279.263585	152.011757	80864.237136	3575.094001	461.358	—	1177.9
102	63.73732	97.00888	1652.763912	57.05748	85.436514	1128.156722	100.213576	3662.879195	9.482275	—	0.020742	0.01983
103	3712.589305	3496.971662	—	4047.902241	687.365672	—	3557.506868	—	5566.685148	—	236.061	31.056
104	150.999029	197.437402	8991.704329	146.525042	122.209519	1841.166484	200.144001	—	36.23162	0.343074	4.65217	4.464
105	8301.681749	1248.941079	—	3000.670894	1129.567495	70261.395572	1264.778544	—	13.784389	—	0.021449	0.020968
106	36.962905	27.474348	606.17575	39.47189	24.850223	255.308462	29.385874	3547.757753	20.487993	0.442435	0.212219	0.193576
107	18.633927	17.111674	18.292596	18.541627	17.894219	18.878602	21.130126	1.122198	226.442828	103.841	33.6333	30.2401
108	5571.493306	3985.122346	—	5074.801554	3279.564679	—	4012.976349	—	14620.621137	752.634	—	519.982
109	3831.618091	9379.168109	—	2075.834368	600.119681	32586.529517	7220.548149	—	25.98138	2.11334	0.463357	0.456898
110	499.509754	140.081913	101182.367188	165.527872	65.61939	849.992115	141.635745	—	12.569111	0.039687	0.029498	0.026287
111	259.299933	242.96312	38910.935711	179.912622	171.014809	3268.821653	246.300756	—	20.728766	0.325898	0.153119	0.116272
112	487.261413	570.306233	107352.816666	452.699598	213.677111	7367.112693	578.484457	—	490.773307	12.3167	103.619	97.5108
113	22.118623	22.435508	186.078021	22.144229	23.80138	144.912995	24.574454	849.793942	46.353218	6.11659	39.7939	34.7442
114	19.13665	163.611853	1041.405653	20.854193	167.289871	1049.629059	32.888811	16844.712381	1190.546909	1580.12	—	—
115	297.066148	377.229723	19197.549417	173.895134	118.870702	2929.455775	383.690216	10302.204231	22.905266	0.331739	0.454338	0.417915
116	861.354386	778.537511	150314.01436	373.663249	231.153973	3188.978204	788.57559	—	8.79038	0.029413	0.026592	0.024282
117	779.23447	297.837472	95597.857206	329.84544	120.76863	3042.642288	301.07599	—	10.003717	—	0.022699	0.023408
118	57.021866	86.066802	9659.270526	62.325888	55.932491	791.552931	90.315529	7704.153361	16.742515	0.137101	0.241702	0.234162
119	128.583575	84.68119	9940.467635	142.462223	67.503788	842.108603	88.058191	14876.120689	12.607812	0.149273	0.200781	0.167646
120	780.024547	425.014274	75744.802002	359.092814	140.46516	4788.025124	430.340828	—	9.49907	—	0.019958	0.019038
121	16845.219713	12414.536787	—	7527.488926	2584.948345	—	14426.848549	—	5281.238882	—	—	933.111
122	145.389294	142.076538	13410.623958	157.879447	147.510876	6667.70499	144.797686	146640.757006	304.891012	183.371	167.302	156.323
123	18.654536	18.906644	18.333149	18.70414	18.608835	18.439813	20.326476	0.927764	69.146296	133.991	595.681	554.832
124	337.112567	164.32999	26257.004541	340.641906	173.654197	4771.955836	170.116025	39247.484851	101.684638	2.8887	1.77849	1.68257
125	121.491269	131.329591	7658.752846	111.428295	117.995217	3115.016603	134.362363	—	68.560814	4.71413	37.645	35.2899
126	438.426931	473.900902	89787.761559	452.954726	348.698839	44980.792734	483.514614	—	11.265864	—	0.020229	0.019794
127	1613.975535	1150.111781	235892.822327	1518.70964	1064.275275	137938.569945	1169.052627	—	387.968754	337.892	2245.33	80.6759
128	31829.59818	29631.213728	—	14178.555172	5785.444621	—	30190.222254	—	240.351362	12.0077	41.0531	36.354
129	2352.590527	2356.334832	—	2694.599671	741.799127	147136.08281	2387.087136	—	13.234903	—	0.02129	0.022209
130	71097.724097	35901.418875	—	21752.828693	5909.179844	—	34667.924477	—	413.682769	34.3672	2346.38	2231.98
131	15002.780451	4862.979144	—	4003.931034	1019.898381	69932.157732	4903.793326	—	15.11553	0.123565	0.129442	0.121692
132	24796.406875	1952.477986	—	2218.763604	1001.377348	19651.527305	1968.872621	280484.940882	17.074775	0.106571	0.088953	0.070032
133	10964.93298	13237.079582	—	6738.222353	6225.930357	—	13427.529473	—	6051.685583	—	—	230.753
134	620.574887	660.083136	30900.357695	460.087208	258.627133	16298.856574	669.068034	—	10.165346	—	0.061596	0.019706
135	1242.598938	1332.775459	236587.362044	1444.045832	367.248423	21344.729627	1348.677018	—	11.106692	—	0.021853	0.02041
136	6001.053822	18237.931926	—	3566.104529	2096.64743	35834.988883	11367.883859	—	106.812425	0.922526	1.31348	1.08859
137	989.581384	85.155756	3803.950753	621.274817	59.140424	1573.189893	89.563425	18900.005367	11.643445	0.41771	0.641293	0.57462
138	2304.170004	1815.608295	—	1004.578679	504.875543	14150.590596	1862.087247	—	14.14547	—	0.022212	0.02241
139	13412.384429	13420.565323	—	3184.361318	2555.459491	50040.824933	13736.478719	—	17.908171	0.05073	0.029933	0.026883
140	539.914422	142.796637	44600.710516	421.973348	95.052639	3836.394705	147.521626	—	13.314192	0.279506	0.323775	0.292326
141	3516.809014	3874.231215	—	3990.04435	765.537721	204149.290453	3931.037596	—	399.823251	170.282	167.868	150.794
142	7296.454947	714.933019	215389.178044	2688.67685	294.098432	25103.985875	726.824927	—	36.638289	0.728397	2.5225	2.49463
143	37.972529	38.525927	153.850877	34.808525	36.505557	134.867152	40.873461	835.691511	9.988008	0.161693	0.091981	0.078048
144	27966.927295	16861.847786	—	12854.555852	5903.250913	—	19063.56082	—	634.916272	6.193	50.6783	42.1672
145	47.993304	125.785269	1677.73974	41.678434	92.25376	578.485928	84.657275	907.498676	22.018499	1.53733	12.2191	12.2351
146	166.051429	876.117856	54266.895673	157.000574	748.908517	3348.472722	301.208357	—	3219.935283	290.504	—	134.176
147	21208.220702	6518.170893	—	6210.137416	1565.489024	—	6568.99684	—	38.370207	4.88413	1.61968	1.59928
148	199.185154	110.539894	20025.186918	214.44767	60.39831	2679.136205	112.707175	—	33.399005	3.13267	1.43285	1.38725
149	24013.489168	5666.422156	—	10352.512364	1306.825694	—	5743.174679	—	69131.784099	—	—	272.703
150	20.56632	19.910056	19.533108	20.076738	19.610498	20.198442	22.042331	1.812514	18.722481	0.048536	0.074867	0.032667
151	28041.197358	528.99075	—	7710.204413	421.888103	189227.524203	536.417356	—	21.619631	0.393173	1.03378	0.982284
152	18.032614	18.282481	17.275853	18.789615	18.836299	18.452254	20.506335	1.091015	18.49791	0.627372	0.424098	0.391162
153	2949.474203	546.216374	29039.802759	813.300805	187.737395	3412.839133	567.118067	—	64.433718	0.780374	91.4189	84.6466
154	366.969251	1850.914998	—	199.295999	1709.45254	74262.020026	695.114622	—	98.734381	204.037	269.418	234.695
155	368.341932	1240.673365	—	333.557052	920.786965	6103.260232	680.618364	—	7675.398026	1027.06	—	101.225
156	775.873023	430.046709	94595.34596	336.262936	199.747052	3732.370661	437.543642	—	16.960628	0.483479	0.198895	0.170974
157	8151.416936	1480.175599	—	7571.702003	1400.16702	282536.00608	1495.168045	—	380.063969	81.1997	957.665	12.5888
158	17.441466	17.518519	17.313773	17.429314	17.554626	18.340532	19.695027	0.694132	531.08381	68.2391	3650.9	3415.62
159	100.910337	42.393191	4955.600488	84.755976	40.833939	734.446088	47.970135	1275.779903	12.980888	0.072724	0.098455	0.071453
160	19065.053257	3442.568251	—	6842.203682	2664.112982	250806.507119	3486.460052	—	16.991297	—	0.019514	0.019178
161	666.75599	223.830624	215.977622	380.384973	262.195544	260.461477	232.734524	—	13.60243	0.152966	0.12032	0.109079
162	1923.814206	645.443888	23223.107347	777.826248	148.94283	7133.038098	656.655932	84490.064136	8.186335	—	0.018768	0.01937
163	91.418574	93.621776	4002.470271	86.432444	63.955365	844.896116	98.880156	76730.903213	18.092836	0.234393	0.281016	0.248638
164	3449.569452	2168.20208	—	3240.24207	2112.182412	36153.246623	2191.567126	—	79.046066	13.4962	43.729	38.1159
165	537.619507	197.711397	9497.294703	391.506359	110.371257	4061.841204	202.635943	70832.857751	10.399112	—	0.024246	0.020616
166	3507.015827	2428.284425	—	1441.709377	463.404999	24977.025673	2466.087043	—	15.911834	0.091524	0.174631	0.128967
167	18.87278	18.740722	19.319656	19.575648	19.602812	19.83193	21.631841	1.452607	12.860181	0.024848	0.023492	0.025731
168	391.136736	352.84631	3082.24085	245.633471	194.836915	5384.436642	358.418262	—	10.216231	—	0.020298	0.022176
169	2854.17705	777.684129	—	1080.160863	709.097947	8352.529165	790.605195	—	14.906544	—	0.021217	0.020223
170	2283.619554	2654.85618	—	1878.878117	1928.761682	—	2706.283598	—	4009.203172	—	—	75.0224
171	9069.797625	651.866637	—	2756.759741	477.120998	109606.328765	683.506337	—	21.017693	0.510611	0.84711	0.718293
172	57.518096	2085.400693	51580.175208	68.909406	2102.913764	43240.390801	102.731577	—	15614.89705	—	—	—
173	1214.716419	611.567403	266945.022844	632.771545	327.359516	34012.112007	619.040676	277482.718818	12.240612	—	0.02263	0.021151
174	8308.346052	115358.000891	—	9027.655561	9871.660201	—	16115.743133	—	1444.195722	38.8027	—	16.5491
175	1028.509683	3104.137109	—	1149.689732	1224.250717	96252.392235	1942.763208	—	1457.09049	28.3087	—	84.4888
176	20.136186	19.820758	19.4811	19.878371	20.088255	19.622621	21.46986	1.6742	15.989687	0.032774	0.066218	0.023463
177	1057.338604	1007.530688	—	801.027055	738.936322	38903.319461	1013.841011	—	106.298004	10.0148	1.31002	1.26222
178	1527.868582	1639.65251	211719.632633	1745.783243	413.733249	117487.840951	1655.170856	—	11.725417	—	0.020252	0.019609
179	2482.932271	2097.842178	276037.981985	1221.854421	773.288133	41366.895014	2123.832582	—	11.845827	—	0.021462	0.018311
180	17.966472	17.532438	17.435862	18.101091	18.534871	18.1131	19.755559	0.980232	3668.359519	540.03	—	124.71
181	1294.02128	8444.712162	—	556.86533	849.099404	7058.788527	2374.450867	—	11.209379	0.069146	0.033598	0.036905
182	49.454472	272.77978	8765.512497	46.925482	201.974875	3365.041491	86.276832	79813.951225	312.613442	50.5668	9644.58	8908.86
183	32.497706	27.461618	140.107789	33.696268	26.153255	76.3018	29.743684	3367.019401	10.943979	0.123385	0.050397	0.046963
184	324.361299	3401.876766	148646.382981	222.174508	3371.943019	101480.204841	625.953886	—	3787.897458	—	—	3625.33
185	431.282186	186.27514	71090.213294	215.245429	188.232123	4132.5866	192.07789	—	10.826791	—	0.060988	0.021568
186	7551.147026	1942.340746	—	3003.024008	639.808915	—	1954.085433	—	670.739117	27.5312	1.70754	1.65895
187	5838.316409	1912.626247	—	2217.712769	1061.514424	111192.686651	1942.066927	—	19.696347	0.147109	0.078575	0.056677
188	720.102182	363.324605	—	549.774788	309.614133	12891.638109	371.169831	—	13.105237	0.128527	0.06451	0.056503
189	18900.688358	9881.900865	—	6527.663867	1998.020841	—	9982.326585	—	72.819883	23.8983	7.43551	6.97946
190	18.736813	20.265614	20.430125	20.059649	20.665224	20.363133	21.685294	1.804504	18.52635	0.022575	0.024386	0.023415
191	1238.616603	294.187679	—	500.160413	208.622828	5988.548855	300.271935	—	14.571156	0.079509	0.053317	0.042964
192	19.406391	18.744142	19.445815	19.243925	19.452849	19.016914	20.927578	1.381499	15.901103	0.035404	0.03118	0.024517
193	48920.088553	12352.967347	—	10796.347611	2291.139035	—	15126.18692	—	18.43698	—	0.035251	0.020561
194	1651.211582	374.629189	—	930.129987	147.13842	112386.813926	430.179153	—	741.185849	43.7145	476.154	10.2637
195	190.863262	702.402251	45843.762838	140.174257	678.239038	36891.23427	360.346722	282002.279551	12.705808	—	0.021531	0.018905
196	228.437	161.41972	174715.266813	203.120205	160.165474	2828.262241	168.074547	—	18.852657	0.11748	0.260071	0.223253
197	1620.975323	264.668199	—	682.444183	262.243062	6751.732277	273.532737	—	13.482545	0.028636	0.026918	0.026236
198	2864.722911	787.927489	—	1361.159495	208.052181	110020.809464	800.033827	—	336.27727	51.8712	3.34697	3.1927
199	759.929632	811.403639	102393.696475	480.015293	252.271533	8381.812769	821.27574	—	14.560449	0.950335	0.208877	0.181454
200	33.483347	354.640726	5239.965074	43.700189	353.236448	1068.935635	57.190684	—	379.966631	7.93166	125.286	111.718

Table 22: Results on *Random Syft 04, 0-100*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	256.495355	228.735753	128248.419483	199.741663	100.188716	13191.074464	242.473378	—	109.559054	1.8342	10.2948	9.92636
2	3161.904989	3587.597951	—	1952.863484	1678.634163	149269.331504	3659.930257	—	12.385282	—	0.020014	0.020004
3	19.457204	19.707575	19.127515	19.056863	19.722886	18.932257	21.901975	1.387975	83.047175	3.64832	3.33221	3.26698
4	3902.566501	1866.096556	32343.601131	1948.857015	876.066626	32705.814231	1909.859299	—	11.884163	—	0.022698	0.021129
5	705.616097	532.830888	—	673.623614	478.865033	25498.68394	547.91626	—	4302.309411	57.2941	872.108	11.7207
6	11681.395936	3889.592045	—	4164.122781	2355.545719	91223.570167	3948.020875	—	15.180614	0.217421	0.101268	0.091401
7	43428.89856	1123.216954	—	15303.335216	655.681661	187814.222779	1150.745076	—	29.932109	0.855873	5.44999	5.41465
8	3510.226511	1814.892139	—	2410.134011	765.912107	128993.539209	1855.342362	—	296.416071	0.63778	92.2801	74.0444
9	18561.571596	8167.49774	—	7213.115722	4559.685007	—	8359.002735	—	402.907789	4.46692	26.8029	24.6481
10	585.236351	1988.598172	—	372.986438	1831.247827	124177.598864	1112.388545	—	14076.898843	—	—	1360.38
11	21640.838058	902.873441	—	6492.992931	830.341252	259001.958349	920.898578	—	67.312154	3.21831	7.35578	7.18099
12	295.844514	221.549517	12177.859371	262.583102	99.31047	1453.648612	228.791538	—	12.336179	0.212305	0.226485	0.225774
13	6352.708638	6748.363224	—	1751.695497	754.30253	116688.418925	6955.648292	—	11.317853	—	0.027144	0.021232
14	2574.308415	1773.096374	72112.663309	1523.058589	555.534069	9062.721652	1834.375203	—	721.031266	34.2325	2161.5	4.16693
15	10322.630248	13619.237179	—	5633.517883	3699.746185	—	11697.118573	—	616.120478	15.5687	785.108	698.9
16	14495.248018	8576.237713	—	5768.296393	3753.506401	—	8804.322921	—	14.921571	—	0.025993	0.022366
17	1956.611661	1062.997874	—	824.412763	300.761052	8041.792541	1095.078238	—	12.60841	0.178765	0.054848	0.046648
18	3746.779149	2551.553243	—	2857.928176	1756.860521	—	2616.422615	—	3922.65084	2700.85	—	697.298
19	372.854129	274.242047	62019.57828	385.769709	208.606032	10605.713224	283.226702	—	407.519899	57.1511	103.476	94.7214
20	1400.936239	198.248605	73464.11795	472.555019	174.086033	3069.904629	203.854643	—	15.597123	0.11146	0.110365	0.101934
21	30400.741307	20266.915803	—	16211.537936	5266.879901	—	20997.093072	—	59796.939283	—	—	—
22	9522.441231	4400.986333	—	6458.327	1828.521081	—	4486.620658	—	13.489721	—	0.021084	0.019722
23	12703.189616	12107.173942	—	3836.640378	2267.930922	—	12461.349779	—	14.055154	—	0.022507	0.019342
24	160.714576	1508.611907	124037.011351	91.442445	1329.384027	12307.857626	304.078415	—	51.709584	72.1763	391.734	346.541
25	350.910966	63.260544	22656.068219	243.806173	52.13708	8423.256812	67.643118	234238.035983	435.350972	3.90005	57.3607	50.417
26	2505.830794	722.947396	48866.713518	873.849846	166.4548	24614.807536	745.100593	—	11.070343	—	0.021353	0.019105
27	1636.772441	644.900088	14582.610336	746.852707	259.846449	12090.767569	663.58866	—	10.146193	—	0.019392	0.018707
28	304.959573	543.91237	81822.763637	229.164734	313.441292	14323.294289	558.487444	—	29.427712	13.6312	3.43246	3.25401
29	7677.310215	12596.668777	—	2359.326042	2186.87236	20943.986241	12968.542619	—	19.109384	0.033943	0.032876	0.029709
30	143.941429	54.865952	7136.127188	141.328147	63.112399	870.313319	58.907722	24285.520015	29.686698	0.97207	1.08895	1.05647
31	860.214332	770.480282	117185.893365	601.157138	580.05095	6245.646097	808.821711	93714.009806	16.656594	0.136608	0.305278	0.313502
32	1471.845552	1739.417779	—	1175.59146	417.570732	43762.387614	1782.069893	—	141.928365	4.27725	62.2051	57.4999
33	298.715068	280.87808	37889.869564	143.153432	188.803014	2922.442182	293.651912	36027.322857	1708.614743	15.0415	—	93.9733
34	852.671306	714.270012	77764.99257	431.040956	196.837018	10588.834775	741.642606	—	9.808593	—	0.05907	0.019135
35	46058.192306	30690.358397	—	27850.263061	6065.580722	—	31814.569792	—	688.584368	82.9573	—	48.59
36	5484.581035	7140.802782	—	2255.000559	1427.846766	195607.312091	7897.744248	—	324.53987	98.2595	148.026	135.819
37	16931.698984	1790.258144	—	6050.406511	598.931518	—	1828.047155	—	63.159424	23.9887	8.57339	7.70389
38	40345.575684	9202.835253	—	15039.814426	2678.669851	—	9484.966396	—	3296.565088	132.1	—	53.0641
39	2879.586744	5412.29202	—	1639.393811	4336.744073	—	5574.991559	—	13.869611	—	0.060323	0.068849
40	60.175214	426.054357	8967.771182	71.579514	398.883474	2532.802191	110.475079	—	1689.829928	543.099	—	6485.26
41	426.874011	154.399056	26707.047181	357.395106	125.839356	4584.21553	162.042943	—	484.756028	168.391	2453.41	2261.6
42	70652.006535	8372.793931	—	21208.310973	2283.442405	—	8630.894312	—	8018.198688	245.895	—	679.983
43	5021.898899	892.610116	—	2839.426622	788.385476	56234.434852	927.367487	—	27.726577	0.973701	4.27502	4.04907
44	607.24779	4642.907021	274646.230552	538.947618	3405.275986	86817.403557	1213.247581	—	31373.785395	—	—	—
45	4952.530003	2303.093205	—	2467.303284	1513.383947	52880.788487	2371.354774	—	15.492841	0.030227	0.062212	0.069998
46	6465.946323	548.042094	—	1597.937487	506.199746	55339.282936	1013.060679	—	16.107588	—	0.020959	0.019752
47	2436.00031	2130.565639	—	1192.156932	557.530247	19431.194507	2183.858709	—	37.692532	1.11006	1.42625	1.30975
48	79.036104	768.193419	9541.763589	74.286927	748.072364	2902.755375	148.348182	—	18788.201758	1751.92	—	1809.09
49	27381.184986	511.748195	—	9288.784952	476.1844	176797.189924	526.054291	—	24.575731	0.304025	2.46142	2.55773
50	762.964759	423.347105	58229.903646	511.450268	256.099827	11618.117981	434.829255	—	349.564814	97.242	234.48	213.993
51	43.706607	226.918651	7783.561492	46.299016	223.818392	4618.506187	79.127728	55016.039077	1688.090983	1571.06	—	2348.04
52	545.826607	500.27075	—	511.168613	414.167775	22813.65781	516.856115	—	9913.266696	771.968	—	1610.24
53	946.242001	419.77742	—	930.913037	413.644407	55675.772157	473.783316	—	932.905653	43.2638	862.296	11.9987
54	6555.3822	1457.664121	—	2655.179329	472.482355	13928.472909	1495.156012	—	27.370011	0.609869	0.50988	0.456074
55	140.651708	1465.500849	152301.491613	109.608593	1427.273021	6968.853126	262.72065	—	51.138669	13.2605	415.987	377.611
56	430.501594	225.202707	91809.110176	296.174438	98.816914	9377.732536	234.799755	—	87.575815	2.63244	16.6731	16.03
57	38154.530475	25440.648478	—	16033.612467	8534.386244	—	32385.001589	—	19071.241342	—	—	6044.77
58	5109.817409	837.204068	—	2428.23398	279.790756	27454.829734	847.448008	—	12.728151	—	0.038354	0.022364
59	184408.901908	40206.413429	—	46665.475949	2443.656735	—	42077.545701	—	139.750365	28.9616	0.764322	0.728475
60	6715.265969	6959.030972	—	3257.053981	1671.037286	243674.719491	7239.651222	—	14.465581	—	0.045812	0.022241
61	2560.977406	1299.134751	—	829.632247	479.715517	10564.607628	1333.706828	—	14.983162	0.075569	0.067469	0.06202
62	2552.273132	5514.672002	—	2377.052485	2588.776135	—	4809.073492	—	48615.857948	—	—	457.687
63	7819.656088	6641.566114	—	3412.835924	3002.412461	—	6798.407953	—	29.09049	2.58196	0.195	0.168749
64	2175.131773	3612.740913	—	2491.675728	1801.647233	216020.807147	4200.061125	—	11.803082	—	0.021967	0.023848
65	30423.272681	5847.340013	—	11729.797207	4135.154099	—	6610.407073	—	17.161422	0.054715	0.03508	0.033636
66	7097.194417	1780.262877	—	2904.985253	1585.81584	171454.921804	1821.08538	—	47.891555	0.323858	5.98375	5.73763
67	2730.294757	1223.137942	—	1399.426156	639.481356	46745.533797	1265.041093	—	23.647681	0.28867	0.929683	0.856437
68	9745.759136	6650.22017	—	2308.611278	1804.573706	41319.615054	6820.157108	—	36.769315	1.21578	0.529795	0.511653
69	453.318253	54.098758	60108.426852	226.929137	54.630291	2541.226408	58.260422	—	13.57836	0.152947	0.116487	0.12667
70	4285.216788	5049.521886	—	2692.228791	3930.374141	217646.817005	5124.318812	—	685.244732	1315.17	1217.24	101.909
71	17.283198	17.206609	17.029712	17.090817	17.127856	17.198905	19.168461	8.75706	10.575735	0.190278	0.321406	0.284155
72	11212.168875	1303.901702	—	9520.056718	292.374888	—	1322.855742	—	19.160316	—	0.023603	0.019499
73	1386.94293	295.546689	107516.962618	795.900041	112.069802	5078.152987	302.410659	—	37.886986	1.01797	1.78337	1.71344
74	204851.994436	19811.027245	—	76490.358871	16930.429067	—	20188.475496	—	26.540738	0.366387	1.44996	1.35187
75	3381.850549	1370.90748	—	1191.193427	325.645417	98339.386391	1389.529018	—	33.088125	0.410552	0.275834	0.224149
76	104561.884928	16400.806487	—	25305.131301	4030.088795	—	16648.207507	—	14.872364	—	0.02058	0.019101
77	235.94869	47.353858	48553.07593	144.615259	58.088661	1184.956861	52.018616	32271.993448	30.051636	2.55769	4.65556	4.4437
78	33915.497321	9462.343289	—	9312.549724	2951.415314	155513.91555	9636.084007	—	17.709029	0.133449	0.084604	0.073914
79	1221.910316	2544.376499	—	673.720617	2028.617143	129982.634572	2328.588344	—	92.66271	26.0946	14.4444	12.4518
80	3305.613687	2189.603846	—	3186.055867	2033.903077	239635.48747	2188.353421	—	3835.46044	160.704	1484.4	1374.71
81	62.811193	243.902846	2339.657754	58.573291	140.103834	860.517883	111.936005	9358.553223	169.568629	22.2561	720.871	626.504
82	551.413718	296.919927	174077.065101	525.470298	167.627414	9430.046591	310.607107	—	39.268418	0.575279	12.8384	12.5712
83	558.602058	536.865052	229394.12745	271.487824	242.513973	1490.105477	552.987154	—	11.441811	0.046025	0.036557	0.035531
84	48224.117965	3820.645545	—	15680.196648	3700.021568	—	3891.608266	—	25.204569	0.334451	0.684068	0.62731
85	14598.407206	1041.678326	—	5010.65728	477.902041	242043.499279	1059.298599	—	17.145407	0.303169	0.296968	0.274118
86	339.776482	56.448837	17081.309678	209.665483	56.605324	839.454358	59.619826	2341.302251	10.501215	0.039226	0.04962	0.04825
87	35770.532875	—	—	13658.329907	33374.408478	—	67324.646632	—	28.401828	0.324969	1.93538	1.82864
88	117735.213	4690.061556	—	40618.332006	1459.910054	—	4785.71739	—	11.627206	—	0.019927	0.018203
89	253.798826	548.703197	15549.848762	177.591971	303.750312	758.180857	463.273446	—	23.933821	0.192361	0.66018	0.613637
90	501.407582	296.428026	284394.055261	578.872189	141.135517	24621.677224	303.959284	—	181.681383	4.7421	520.218	484.833
91	9619.504757	2166.989565	—	4087.726677	1311.891118	49623.231015	2193.163477	—	15.143075	0.039694	0.026211	0.027755
92	23464.466904	6824.843367	—	10160.221937	1853.120632	—	6816.20311	—	84.802205	0.68981	0.842845	0.795024
93	25895.790943	10783.938121	—	12499.787349	4533.769446	—	10899.066962	—	14.402145	—	0.022718	0.018492
94	18435.027887	2204.622579	—	15518.824543	2101.149577	263802.946589	2231.349286	—	31.912914	7.67057	1.24115	1.23357
95	557.80243	204.012645	46249.636139	543.086807	208.257887	31339.989397	208.022781	69479.127892	13663.433513	361.839	—	153.594
96	218.430251	235.452514	10008.486475	236.493684	236.320827	3244.6971	242.555626	—	5980.444998	1780.23	—	356.269
97	2351.318509	339.792965	—	1125.268508	192.915219	38489.248804	348.018991	—	16.312196	0.265708	0.115217	0.091023
98	17.181187	18.202233	17.281317	17.567556	17.697807	17.730045	20.120973	923.791519	34.412129	2.69854	2.42289	2.15574
99	1076.335633	119.126789	21834.512141	814.452627	102.248913	8211.655204	123.87565	54404.231159	181.445709	79.7902	1370.06	1302.36
100	112.036157	748.602877	22969.721018	100.443287	750.211039	7591.321423	206.283515	—	68.283713	142.29	597.963	527.933

Table 23: Results on *Random Syft 04, 100-200*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
101	295.002645	261.686271	30830.986106	242.378586	123.063882	2462.101056	266.229504	—	12.177656	0.090917	0.111282	0.101919
102	1942.392116	3524.821693	—	1339.39268	2501.494184	79674.19967	3554.085578	—	40.152494	0.74228	1.00443	0.96298
103	25135.384243	978.008038	—	8649.644533	436.282311	136539.597599	1000.435108	—	12.680067	0.076829	0.086768	0.088021
104	23207.335679	2252.594011	—	8433.401777	2147.376905	149511.378653	2280.106977	—	37.502715	0.734245	1.03674	1.02668
105	63682.554619	3183.390487	—	26685.771334	1057.304424	—	3239.980873	—	60.325145	1.85618	85.7931	81.7416
106	1528.021637	879.138803	—	1445.727308	783.596915	53950.231467	888.384367	—	22661.341436	—	—	241.9
107	18.541823	18.994679	19.118518	19.186762	19.22695	19.266799	21.059284	1.312632	19.04481	0.16584	0.357022	0.288947
108	24262.05354	7697.267922	—	4972.409866	1848.635115	—	7799.256021	—	130.895069	74.1143	547.103	68.8497
109	197203.706206	56351.121265	—	39786.930291	12419.175767	—	58086.636578	—	26.63248	0.517648	0.976711	0.871435
110	50.082337	264.174597	24886.083543	58.419485	259.100972	7363.12572	89.49381	—	6482.320884	382.08	—	364.915
111	1579.76613	462.323063	97875.190785	1218.169889	164.567686	39998.376153	467.729562	—	11.485207	—	0.048157	0.019753
112	167099.02908	2136.910782	—	42013.970832	2089.409793	—	2161.005926	—	25.909877	0.135875	0.269614	0.207768
113	84.956232	525.652097	18389.070589	96.238876	509.640311	6145.618071	150.830328	—	7738.505539	418.437	—	642.242
114	6091.849202	1495.339359	—	2414.540173	1416.604272	173931.383106	1505.355092	—	16.324846	0.084881	0.084213	0.064998
115	16915.802909	4140.352726	—	7877.185978	2575.813298	—	4177.237597	—	41.450981	0.459932	1.11997	1.13353
116	17.282493	18.61944	18.196442	18.475435	18.543628	18.409714	20.466597	1.020007	86.88797	19.731	11.8289	11.5422
117	117702.942111	25177.54123	—	29501.158199	11644.742782	—	25653.310888	—	44.421799	2.93585	0.243729	0.22557
118	602.014842	146.318755	53125.708309	430.626298	57.9498	2347.611035	149.803422	—	35.454966	0.565157	0.509263	0.467137
119	17079.444674	1471.369008	—	6223.601892	1080.840964	—	1490.670282	—	15.86605	—	0.020507	0.019178
120	18.955499	19.537727	19.270884	19.598377	19.670142	19.379838	21.320693	1.401701	17.147802	0.115703	0.069775	0.060203
121	9265.312985	7965.752086	—	5409.57241	2310.343264	294592.742179	8042.481043	—	13.023112	—	0.020921	0.018864
122	2703.009424	1360.21951	—	1058.544336	705.511888	51046.116116	1379.470458	—	9.980574	—	0.022256	0.019012
123	139572.133038	6218.677511	—	39699.311122	2320.879234	—	6302.122724	—	116.145501	4.26201	5.21081	4.90183
124	319.48953	3717.73629	—	389.061058	3527.580257	—	593.813405	—	—	—	—	—
125	17424.2287	1549.973828	—	8546.897067	368.290509	130224.355003	1568.977772	—	220.092785	4.71063	90.6397	74.0987
126	52571.47372	11580.625944	—	18090.31358	2704.415433	—	11684.770171	—	9545.561496	532.105	—	79.2619
127	60.518512	2070.657135	61085.816963	64.483211	2035.799845	21565.161707	110.86543	—	699.502149	878.982	—	1898.56
128	22910.905951	9094.130882	—	9186.557544	4124.879185	—	9235.317173	—	13.530057	0.037198	0.07001	0.032894
129	24787.021745	21629.558546	—	12100.52466	4071.295051	—	21024.230335	—	17.493794	0.092462	0.057589	0.054558
130	5623.727709	2561.621739	—	6002.697221	588.69449	—	2609.144744	—	16.204276	—	0.023202	0.026659
131	4790.6067	5206.654275	—	3393.613734	1287.471214	—	5296.01174	—	13.40788	—	0.021512	0.020052
132	3992.133373	2440.945546	—	3730.590643	2390.775487	144271.035437	2481.310835	—	363.458704	20.9587	153.353	147.208
133	184.303817	496.893772	43217.242535	170.88474	487.007179	15470.87304	341.12779	—	11.856997	—	0.020322	0.020416
134	12352.057701	3655.752288	—	3920.05494	2325.938333	—	3684.270456	—	214.382606	30.5838	183.033	11.0918
135	6289.708245	2639.103669	—	3410.144308	1298.161607	250821.632004	2664.743298	—	218.009215	1.98704	69.7875	1.23422
136	256.211707	1139.232271	98479.552039	206.094718	1089.726006	14802.157888	475.369726	—	47.811135	36.3718	52.1515	50.2812
137	19.453933	19.707749	18.934758	19.458831	19.642882	19.978092	21.997288	—	15.367699	0.141069	0.073359	0.072122
138	16815.550814	4720.687999	—	8638.758685	2178.328893	—	4783.394774	—	5854.59995	—	—	405.688
139	35498.115175	1493.044072	—	11895.014781	1171.86572	—	1503.638438	—	14.278959	0.103692	0.089409	0.085838
140	16821.691221	2953.515292	—	6587.715603	1881.061169	198266.956883	2982.6445	—	16.223891	0.031404	0.027116	0.030963
141	6550.288913	630.648611	—	2091.073347	696.161092	48084.631618	635.830667	—	18.945387	0.042142	0.031366	0.02829
142	4046.17785	4790.711061	—	2203.784237	1668.798315	—	4861.442726	—	37112.983704	355.542	—	75.8374
143	116810.158866	15652.762767	—	46349.989841	3405.663127	—	21549.940115	—	21.381831	0.164899	0.705026	0.615957
144	1093.643638	428.606985	—	503.74094	385.914467	27755.736725	432.951567	—	10.696694	—	0.020671	0.02306
145	19.147668	19.645787	18.817636	19.300472	18.998961	19.244106	21.087269	1.334178	62.627937	3.16025	2.4306	2.3327
146	596.687497	2048.512342	143970.172238	434.075953	1773.402605	83009.675406	1132.012702	—	12.856968	—	0.025382	0.019254
147	476.092004	205.49364	—	275.264535	201.328699	23646.78278	208.918333	64373.087406	463.148487	1030.89	—	825.419
148	8267.73296	1408.27176	—	6938.954457	398.970241	133585.942119	1414.349205	—	121.640037	5.94517	40.3145	33.6573
149	12703.085285	12786.142046	—	11695.831044	2301.338661	—	12939.697435	—	4152.334142	350.136	—	321.677
150	674.479855	1415.253534	24592.167741	465.828788	1141.648771	21754.378169	1311.809388	33846.066146	10.316114	—	0.05971	0.020556
151	2181.17635	983.572895	158225.9424	1030.917766	732.197414	132545.847861	994.385964	—	10.342336	—	0.020053	0.021989
152	3422.222921	2065.500407	—	2203.477228	861.6786	264097.323057	2075.385528	—	108.333895	6.05827	96.3105	88.792
153	11051.781937	2995.061018	—	4889.097496	689.779356	213956.809484	3020.871961	—	93.19003	40.855	92.1418	81.679
154	7350.8738	5994.397241	—	2046.313593	1625.285211	111954.872938	6072.812831	—	43.464869	14.386	1.80358	1.70131
155	630.878605	1434.256944	61097.354375	260.58163	422.529703	29008.702219	1189.232885	—	10.042678	—	0.021167	0.019553
156	4777.601307	2663.464851	—	3770.119118	2480.609088	92888.409152	2700.78867	—	64.062402	35.3638	114.989	118.627
157	8799.21745	2949.890227	—	3958.354953	908.512862	—	2985.894889	—	2371.337117	582.804	—	173.435
158	8651.899577	4760.178225	—	5735.078816	1465.182022	—	4812.289519	—	9035.162744	474.513	—	93.0846
159	30700.123624	6792.93122	—	10832.374287	1332.208031	—	10713.482818	—	11.959961	—	0.060955	0.022265
160	4187.306799	609.765441	—	4151.230371	626.478276	—	621.404598	—	13.217071	—	0.019735	0.019924
161	35752.029033	2397.738167	—	10501.438415	965.398373	—	2401.850493	—	20.017663	0.210468	0.322652	0.255714
162	476.434784	173.974362	196415.212529	222.569046	168.717822	7793.698766	176.004768	—	10.042409	—	0.022035	0.025584
163	20.472289	19.199831	20.124682	20.6493	20.662631	19.763718	22.713233	1.71419	15.810154	0.033012	0.026127	0.022262
164	28738.439163	17327.06059	—	10136.34161	2808.211991	—	17779.765329	—	90.975572	112.078	153.292	119.955
165	1293.896615	542.767309	255664.490273	820.345445	344.933072	16399.697217	557.907149	—	67.882841	0.695871	10.2986	9.79484
166	18.797875	19.170747	18.969773	18.820475	19.015777	18.372774	21.391502	—	12.666223	0.135254	0.065085	0.055151
167	761.724429	745.632744	—	705.302424	654.222243	137453.073077	751.55452	—	11.578137	—	0.022109	0.018187
168	103981.828138	71002.675576	—	31920.826801	10766.765169	—	74539.621223	—	11423.539253	—	—	229.065
169	46748.251531	19936.433818	—	10142.618523	8682.639277	—	26039.632139	—	45.373596	17.7569	9.07006	9.12449
170	3195.758935	2119.029385	—	1698.806984	839.399202	101475.763322	2152.517056	—	10.409174	—	0.021295	0.020485
171	4390.479444	908.110932	—	2083.821806	263.951424	48843.569094	917.1012	—	19.328179	3.91127	0.795724	0.759036
172	3018.587193	447.562838	—	2781.182951	428.230118	180800.522092	467.056332	—	1620.941125	3449.99	—	222.775
173	4323.845551	3451.304938	—	3102.469169	2505.969239	—	3525.395448	—	43.282675	1.68053	3.08216	2.91662
174	2554.051742	5907.767032	—	1196.621783	2576.038959	139990.650503	4876.77493	—	1363.115145	94.4761	905.789	829.57
175	2471.758755	4327.499158	—	1831.212644	1943.785709	196991.401568	4378.646977	—	180.603681	90.8907	77.7662	69.1643
176	19.865848	18.839598	19.596072	20.103318	19.53977	19.467566	21.53432	1.300119	13.695123	0.028363	0.028682	0.026498
177	6500.979661	1259.912319	—	5443.895723	1084.504709	—	1276.67141	—	5150.220849	193.904	—	95.3437
178	561.295307	543.481012	—	500.704613	422.251382	17648.330945	550.065898	—	66.562657	3.82842	14.4919	14.4371
179	12227.884017	7500.89873	—	8040.709637	2493.761636	—	7716.080484	—	720.919641	18.2298	151.109	132.967
180	305.329109	536.691927	15664.405887	290.869207	531.529132	10415.753021	559.664875	18703.744824	2955.029202	206.975	—	55.1586
181	1311.134414	667.147396	—	876.064164	348.303538	16812.1785	698.838055	—	42.77211	2.49336	9.97563	9.60647
182	234119.38395	24920.780063	—	56795.741359	8125.502434	—	25875.488425	—	566.940447	13.7887	58.7461	47.4498
183	3216.729419	319.68414	—	924.007274	314.526397	10860.947504	331.851185	—	18.558792	0.303399	0.367643	0.2938
184	977.71116	870.500028	—	565.985504	320.484067	12470.763103	916.142455	—	15.33436	0.142309	0.267633	0.262873
185	2611.133769	558.086736	—	696.441045	329.767034	13745.176184	583.152713	—	17.656473	0.566485	0.210732	0.220126
186	3551.764252	4414.804474	—	2291.027571	2921.029929	—	4519.85677	—	1657.784013	15.1741	504.296	465.859
187	11883.240069	5535.203745	—	3764.475909	1277.508867	—	5764.441676	—	1333.579374	55.1555	—	81.4687
188	16031.468044	14109.903349	—	6342.15473	1425.955311	298194.309857	14654.467736	—	11.278159	—	0.065107	0.020588
189	24202.08847	4518.47586	—	8748.074767	3083.013263	176310.665213	4629.304288	—	16.560549	0.0487	0.033035	0.028518
190	401.430445	332.571845	—	410.565524	191.922243	22563.057961	348.102298	—	19.624248	0.132543	0.269362	0.238565
191	17.99	18.33202	18.343717	17.765818	17.791808	18.420414	21.365875	6327.938278	630.189379	381.79	4020.12	80.2074
192	18.322788	17.849942	17.71427	19.024363	18.497113	19.554474	22.529594	1.286493	13.532845	0.04602	0.038673	0.037435
193	274.703485	243.461927	72269.990289	190.072043	223.093316	20793.015464	254.149432	—	15917.123513	7151.25	—	783.317
194	1477.115521	447.143557	—	488.422108	288.533923	3198.066257	461.351907	168353.322414	13.522819	0.048428	0.035336	0.028617
195	1052.286855	685.423775	—	722.774317	426.093266	97064.527392	711.688533	—	1644.659012	249.529	1693.39	32.9728
196	2643.864153	83.84095	—	696.859054	95.896096	4229.511334	155.505308	281490.192442	14.915795	0.053033	0.058248	0.052904
197	52433.734468	6750.484987	—	13315.076	5586.49299	—	6931.634893	—	30.371204	0.393057	0.265987	0.214521
198	318.52136	467.072126	34942.91639	334.422027	259.692389	5388.592068	489.753125	—	95.885999	0.821162	56.2739	52.4707
199	126182.092378	10526.250212	—	19141.218811	9799.531847	—	11037.944727	—	82.854819	0.532961	12.5162	12.151
200	3548.378746	4265.277558	—	2214.36371	1013.606915	—	4423.322452	—	13.532977	—	0.019796	0.024898

Table 24: Results on *Random Syft 05, 0-100*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
1	58700.330788	2990.400872	—	20294.760652	2017.257092	—	3045.018323	—	73.348249	4.43764	36.4456	33.1644
2	141128.69629	137284.051836	—	28655.121064	27220.550187	—	139290.802305	—	17.210375	0.169509	0.080361	0.069687
3	8940.169389	3308.628316	—	4689.274945	2791.597485	188405.001144	3311.865263	—	35.006535	0.731498	2.36933	2.20971
4	7816.447644	767.415549	—	5432.115451	590.001574	284741.627653	771.415557	—	1414.800526	858.889	—	220.26
5	2482.345045	169.243278	—	2411.419836	167.680277	74869.149956	172.670223	—	28740.409431	693.476	—	1802.5
6	1229.462051	824.254725	—	1252.748591	594.600642	—	828.221929	—	10742.37776	45.1448	—	53.5212
7	422.816407	259.31367	79747.567677	390.160987	251.17055	34541.595055	260.90415	—	9.664667	—	0.061738	0.019271
8	290.270328	354.605701	—	237.964082	320.252766	75835.495815	356.524559	—	2630.081081	—	—	749.62
9	908.579959	603.950842	147091.400727	589.570175	243.312028	5298.841251	607.968146	—	14.228571	0.231575	0.437286	0.42008
10	2776.014956	3377.356946	—	1400.294562	3215.573188	66568.461546	3400.852308	—	11.540968	—	0.020551	0.020436
11	4561.561287	4906.646676	—	3352.378531	1803.425981	—	4924.511546	—	12.086678	—	0.023772	0.022296
12	1206.188263	216.460045	—	705.984161	179.511497	29802.935194	219.615799	—	9.986439	—	0.021042	0.020071
13	22060.570647	1518.665733	—	14760.616557	1044.758461	—	1520.955629	—	8.961653	—	0.020674	0.019625
14	19586.732912	3224.752069	—	10985.168243	1553.940856	—	3236.041756	—	9.599222	—	0.019806	0.018413
15	174.860352	918.163914	—	163.747455	543.292246	28674.405703	315.386866	—	1346.481302	74.7925	2965.7	51.4804
16	122368.949913	18059.210094	—	55478.308181	2931.429951	—	18195.075795	—	632.596444	3.33325	1050.69	4.04155
17	1008.058108	318.849241	—	601.284426	298.682834	66858.256916	322.750051	—	3973.69759	—	—	1505.81
18	27305.527748	16680.791168	—	13809.879371	2068.116533	—	16783.145669	—	159.252598	1.77571	38.5818	35.2867
19	1127.899574	1013.192438	—	484.389554	985.910701	35242.964355	1015.414729	—	12.971369	—	0.022197	0.024832
20	28004.995636	6012.750736	—	15441.358207	4247.583707	—	6028.904249	—	616.88824	36.0997	2924.81	38.1516
21	46995.20179	15958.9032	—	18143.878689	5002.276841	—	16015.867739	—	10.256214	—	0.021404	0.020474
22	41939.801868	56589.627643	—	11846.930374	8095.387802	—	45843.177392	—	4054.44576	19.9354	—	28.678
23	2276.962163	2407.516481	—	2042.815485	1881.321586	—	2423.523242	—	27181.838151	—	—	2432.9
24	2382.965078	150.289695	—	1312.397557	86.161235	35881.13416	152.018178	—	11.552534	—	0.049887	0.023302
25	342.579796	1055.256108	32919.480535	150.057803	652.567809	7475.080215	628.757361	—	278.981324	194.144	—	609.615
26	2548.49366	1057.666482	—	2046.095145	653.284112	—	1062.23481	—	11.026902	—	0.021623	0.02014
27	5681.003327	8478.06195	—	2722.518862	1533.487194	21023.226816	8490.91359	—	16.870401	0.644369	0.208556	0.197683
28	22779.023815	15780.579909	—	16747.897381	9719.848465	—	15798.1331	—	12084.753991	11259.2	—	18164.3
29	323.438963	188.702875	31309.826172	273.155515	114.311822	8315.587015	192.307801	20092.519	9.016868	—	0.058624	0.020551
30	8111.839713	1089.346128	—	4508.981958	336.23335	167490.471467	1094.754183	—	12.799015	—	0.021452	0.019478
31	30220.744632	10772.380292	—	8550.208048	2107.530857	74724.852915	10796.916829	—	14.168532	—	0.021149	0.019517
32	36026.550087	2923.667522	—	18535.184301	1593.114583	—	2934.939404	—	52.028249	2.20089	5.32894	5.17455
33	5457.047566	4144.309633	—	4144.972211	861.48506	245790.614812	4201.145213	—	404.246266	17.512	82.7551	69.3767
34	10278.205226	4312.48224	—	5566.168166	2269.719957	211960.171427	4324.788415	—	14.785487	0.032269	0.027274	0.026882
35	29795.082667	16260.560115	—	12501.806856	4632.845408	133478.318231	16331.157336	—	42.417759	0.204222	2.38687	2.39425
36	10331.827851	10601.895384	168702.048019	6129.331877	5865.042462	139313.537573	10607.595991	—	11.474357	—	0.020245	0.019813
37	2725.911541	1275.001065	—	1501.958759	1043.543016	198886.785053	1267.759559	—	194.62376	78.4146	858.345	769.508
38	151.14027	101.48364	6203.264951	143.364133	101.895832	619.672523	103.271318	4055.895728	792.038779	276.011	4817.98	4416.24
39	10768.331761	597.169816	—	4553.003204	299.064568	108546.654545	607.762678	—	198.036013	5.5938	1334.12	1245.71
40	46887.543055	50794.434161	—	29139.970488	2859.884197	—	51436.611373	—	13.233128	—	0.021302	0.021823
41	8627.047396	7349.92334	—	8468.252463	4245.584072	—	7372.088139	—	82575.408922	—	—	—
42	32694.169314	1346.249937	—	11079.105089	1258.003165	130516.550047	1357.925349	—	21.493095	0.188134	0.448735	0.320219
43	66221.166189	17787.337552	—	27106.468983	2060.675577	—	17878.23406	—	15.449197	—	0.02166	0.020349
44	82040.188355	5049.051541	—	24891.550338	1920.190858	—	5049.454603	—	116.725539	9.78349	167.529	141.7
45	111589.322432	19715.695244	—	31259.874217	6252.870173	—	19766.3226	—	231.713301	28.738	75.8915	69.0689
46	6877.275047	5198.476979	—	4232.935669	2350.083289	—	5186.10696	—	5718.881089	234.362	—	156.146
47	1685.500136	367.722837	—	1291.805795	130.009063	—	371.84889	—	11.072001	—	0.020869	0.019589
48	38201.483738	8684.741989	—	14298.603306	3356.14556	—	8818.640246	—	19.443872	2.81047	0.445213	0.422222
49	30446.621776	16071.431927	—	21110.714254	6824.289215	—	16158.411535	—	55.073562	1.66613	10.1282	9.56761
50	808.702679	2004.775718	—	648.176473	2000.257277	147505.587691	1508.653032	—	2814.492087	—	—	3254.15
51	79870.284468	27039.497935	—	28341.504227	3875.765594	—	27216.39657	—	23298.760913	428.082	—	447.735
52	—	4297.204216	—	142763.813753	975.165527	—	4316.300418	—	1903.440879	54.5587	2696.27	15.8142
53	2293.419618	1687.627417	—	2291.491187	1686.368274	166153.53834	1688.38159	—	19366.810394	—	—	—
54	6391.245161	7280.390148	—	3870.921949	1695.947631	265493.15171	7280.450564	—	13.973014	0.214685	0.103811	0.07674
55	41890.690649	299.62048	—	17169.560463	182.764561	125783.221333	303.983505	—	48.177516	0.286914	5.05796	4.92432
56	70538.594124	3083.870253	—	29113.213469	1135.339266	—	2959.924875	—	11.553241	—	0.020626	0.019988
57	19.96096	19.663134	20.167354	20.38119	20.184889	20.398332	21.745433	—	14.614414	0.061915	0.040828	0.042583
58	1052.283541	897.001044	171692.45186	854.883303	500.141605	13833.436406	909.509701	—	10.743315	—	0.042028	0.020303
59	12103.469956	1626.941118	—	6486.263538	854.672909	—	1629.526773	—	198.944376	2.70413	50.5322	44.1135
60	1877.445495	2835.500481	—	1535.463864	2368.999583	—	2838.625823	—	10659.146149	246.904	—	87.2884
61	1207.253726	323.799285	—	1061.680873	324.814942	47826.042231	327.053373	—	2891.166817	873.302	—	244.018
62	715.052257	269.464349	62809.999408	479.872075	188.08073	23295.21443	270.982204	—	136.24147	17.505	570.453	29.0511
63	135318.842367	26403.236274	—	45417.539051	6533.789692	—	26712.683863	—	13.587084	—	0.021591	0.020909
64	39346.063752	2485.955963	—	19287.34551	554.709275	—	2510.027457	—	15337.764082	—	—	1889.33
65	2230.870848	1136.039995	—	1790.012536	1075.328145	4209.106976	1152.868329	—	15.699673	0.387254	0.249575	0.250466
66	405.04787	979.864949	—	280.006363	661.224271	46935.080583	746.973118	—	333.008792	10.624	27.3626	23.2703
67	23563.631679	4462.102163	—	11394.36493	1336.849044	—	4458.088495	—	31654.511342	—	—	1642.34
68	19.137556	18.61372	19.375038	19.856665	19.357889	19.261383	21.283478	—	24.087823	2.2532	1.07976	1.02903
69	2623.459003	4672.141306	—	1523.907467	3074.266612	—	4694.932329	—	13.129133	—	0.022448	0.020783
70	10252.129892	1638.215931	—	4715.949896	681.31875	161997.663676	1645.666907	—	41.683465	1.77815	4.3974	4.213
71	55.854341	193.831575	744.138934	56.344255	194.410901	3257.806619	101.70087	2445.032522	166.292418	224.294	2328.29	2065.11
72	81469.948613	7076.204303	—	13385.017603	890.253417	—	7083.404944	—	11.163133	—	0.024193	0.020281
73	104.099392	70.194601	25659.842229	97.542763	63.839856	12905.379983	72.775513	27915.052305	2311.45297	12.8508	—	21.3147
74	6333.82123	522.351215	—	4436.80528	263.386138	—	525.805495	—	5849.638932	187.999	—	31.954
75	1109.924664	235.5938	149051.428826	776.70092	130.340633	50609.526081	237.660171	283769.02714	9.74164	—	0.054111	0.02058
76	154827.791405	7583.734721	—	57633.900832	2578.551598	—	7628.31978	—	100.489029	2.89343	34.7436	28.9774
77	9524.166996	7761.014994	—	6596.999382	2833.230246	—	7750.048458	—	—	—	—	—
78	1896.166383	934.774299	26081.540162	1755.502308	893.230147	10234.218341	936.172934	—	11.129078	—	0.021221	0.020332
79	9711.013564	8928.415015	—	3858.527317	942.80562	52950.984201	8971.000315	—	9.902823	—	0.027898	0.023044
80	11414.292399	6420.406263	—	5584.70324	552.449801	—	6457.433405	—	11.581165	—	0.026891	0.025769
81	104248.924017	8062.036524	—	16204.706095	4233.852955	—	8076.96556	—	33.713198	0.9188	6.36559	6.51653
82	16148.503649	17401.431254	—	8825.756909	3174.721273	—	17578.514428	—	12.773708	—	0.027597	0.021661
83	151.502919	499.841963	21824.110788	117.074323	477.12128	1436.756718	275.028057	265869.54279	214.83311	7.75776	82.5587	60.3383
84	24408.673719	5268.732659	—	7321.066317	2323.373755	—	7795.132989	—	64.041727	16.4526	11.1158	10.283
85	19449.705986	12608.035344	—	10157.237636	3441.836509	—	12666.349474	—	11.732479	—	0.021678	0.019837
86	268582.842546	15272.160856	—	108132.628358	4497.444776	—	15422.023341	—	—	—	—	—
87	1124.932541	129.495996	—	834.041661	104.129539	24909.895721	132.871182	230702.111015	585.90562	34.5555	4974.22	31.297
88	11720.92758	7731.704982	—	3676.198361	1315.636344	—	7816.531022	—	11.072075	—	0.021727	0.024843
89	12659.242716	3399.049935	—	7470.936161	2301.451173	—	3414.104743	—	2577.376869	33.3131	5864.11	16.8167
90	1012.743374	2232.924155	—	602.362741	2006.712506	32616.846822	1865.147023	—	752.755136	166.417	5937.47	34.0897
91	532.064928	315.167288	14020.324527	499.082727	243.183486	6282.169072	319.193611	19157.683312	8.968402	—	0.020456	0.019761
92	7055.653164	8030.383992	—	2830.56065	3112.085804	—	8088.065967	—	428.099606	2643.19	—	20.4715
93	18875.080678	8229.263249	—	6349.615722	772.757983	119278.488206	8295.266474	—	59.92052	1.61087	3.65878	3.52094
94	346.627805	294.045201	32388.947096	191.77103	258.930113	3273.747398	296.816973	—	63.038296	1.97943	11.4597	10.5733
95	—	96202.068731	—	63470.438626	14190.586011	—	96779.071536	—	45.458503	15.3124	4.55942	4.67501
96	19.412523	18.813308	18.623944	18.770261	19.151119	18.584323	20.592042	1.250781	14.220639	0.514663	1.26755	1.13969
97	2854.924009	1104.945418	—	1265.8904	386.342752	77247.25626	1111.444406	—	152.817674	31.71	1465.48	20.6349
98	16408.669515	1585.576443	—	5932.684528	833.946243	200281.461663	1594.466049	—	11.369562	—	0.022176	0.027706
99	18.836146	18.599859	18.832898	19.545638	18.894813	19.039171	20.743523	118576.973242	25.291991	1.10615	2.90432	2.73659
100	3470.298417	460.761485	—	2153.914767	309.9182	219792.501808	462.929537	—	1749.585741	171.877	—	98.1923

Table 25: Results on *Random Syft 05, 100-200*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
101	11698.55439	3417.715229	—	5215.458081	680.807943	—	3413.549859	—	39.664408	24.8839	48.0834	46.7791
102	38812.320531	3398.762652	—	19117.586499	1162.067416	—	3408.450807	—	30244.842227	—	—	1947.48
103	89455.335706	27094.714908	—	43400.077024	22596.117104	—	43730.816842	—	842.856736	6.19327	218.777	3.72073
104	11103.102123	5989.265511	—	4251.804433	2166.598487	187442.459665	6022.333877	—	11.986854	—	0.021689	0.02161
105	20286.139818	1345.61415	—	8148.018682	403.962658	98424.359682	1352.164198	—	12.967107	—	0.021439	0.021326
106	16699.783977	3311.536535	—	7725.362959	1126.600651	—	3305.053633	—	12.411959	—	0.023527	0.023249
107	213347.76552	138242.852168	—	76715.798005	18652.59465	—	140503.700835	—	57.010477	11.235	2.03394	1.87405
108	37379.861511	24242.199284	—	10768.730384	4317.157695	—	24370.222851	—	12.340275	—	0.021646	0.019075
109	34919.590209	4893.927256	—	20414.760103	1196.280984	—	4888.337402	—	6251.418735	718.626	—	975.2
110	127274.712997	9584.722123	—	26055.946656	1832.638488	—	9646.546104	—	49.856221	9.85105	15.3067	12.8829
111	62654.81867	14268.550994	—	20673.665349	8868.764681	—	14319.039852	—	11.76393	—	0.021315	0.019446
112	204897.717379	34275.668705	—	51318.292112	6179.441947	—	34469.069728	—	13.111383	—	0.021454	0.022389
113	29276.232097	14951.892498	—	13130.224497	2563.746633	—	15001.396985	—	23.829805	0.313981	0.485369	0.409485
114	260459.395871	117222.342695	—	50601.462334	25731.154384	—	117992.747588	—	16.87594	0.239269	0.138055	0.110339
115	—	19435.472377	—	—	5419.186899	—	19479.941008	—	713.907361	27.0144	—	250.435
116	8016.252009	2411.984107	—	4337.650503	608.95524	44479.36879	2438.980802	—	45.906513	1.30291	1.76374	1.67993
117	5415.794338	3477.177237	—	3454.673287	2810.833085	41365.508755	3481.801287	—	12.337808	—	0.027947	0.020899
118	926.798734	2273.529582	—	472.592308	813.44719	17064.674642	1724.392599	—	25.760802	0.129766	0.434462	0.413395
119	19633.063539	3043.205176	—	15429.878608	2104.207168	—	3048.920054	—	62.440265	17.9433	42.0292	39.8836
120	123890.115259	—	—	32624.539743	35559.895736	—	227395.141141	—	90.290839	3.89388	154.391	147.117
121	17.976393	17.907602	18.308356	17.737155	18.581091	19.105432	19.634738	6906.797893	24.465386	4.05426	7.05405	6.68111
122	2926.261851	1272.765583	—	1264.353596	504.32085	171047.37503	1277.563604	—	12.647077	—	0.023142	0.019967
123	132.561924	285.801252	—	117.642364	158.19171	6562.748673	238.077212	—	83.530345	3.69884	6.15247	5.98438
124	13895.74016	584.947226	—	5972.004586	491.614701	—	585.971491	—	1268.93645	226.73	—	1312.14
125	81.62161	135.043226	2388.90184	75.919923	114.479745	1145.143475	133.497193	6584.834142	64.261245	4.51852	102.304	83.5576
126	25670.968441	17045.18735	—	10840.184125	4767.733409	—	17133.738198	—	55.300621	7.30955	32.1783	28.4759
127	503.846829	761.476555	—	365.026304	746.949403	25572.128439	763.743618	—	143.529892	34.2626	284.185	241.682
128	65282.368326	8528.73778	—	19085.431186	5533.682852	—	8585.816843	—	16.615518	0.220529	0.186843	0.132243
129	3475.584441	1718.890178	—	2687.41861	716.856508	—	1724.621994	—	1919.866972	28.4374	—	23.2328
130	1920.651763	1459.035194	—	1266.38669	304.361094	14510.288623	1464.738436	—	229.873673	3.16496	62.3088	53.1857
131	28803.177623	3716.207829	—	16040.519504	2745.926607	—	3721.415704	—	28.516968	0.171525	0.810644	0.729634
132	6982.127929	277.659596	29245.751748	3670.645915	155.755668	19338.778853	280.850886	22745.665849	8.188213	—	0.024829	0.021838
133	11533.961689	7729.390851	—	5599.319222	1317.024384	—	7715.224484	—	55.164555	121.757	3.78245	3.56273
134	76477.77149	9232.000625	—	33942.513257	2553.410682	—	11283.147045	—	—	—	—	—
135	149913.475332	14103.879815	—	56214.754959	3637.763005	—	14208.966005	—	16.744235	—	0.027078	0.06088
136	1362.274579	1453.398868	82153.226339	901.293772	359.559811	23581.888578	1454.297187	—	9.43073	—	0.020274	0.021511
137	5899.774576	3637.136383	—	3628.040538	1237.998927	105026.109279	3655.057386	—	25.381909	0.191452	1.0759	1.03524
138	103261.647691	156607.756849	—	25240.347092	20111.775331	—	157546.349265	—	30.135673	1.77365	5.16541	6.00157
139	5811.175079	669.831736	—	2631.543265	131.628467	—	673.132339	—	255.973672	59.4978	19.1027	14.3506
140	2744.107213	1494.222127	238569.468982	1937.572979	470.751153	41614.311706	1877.682649	—	11.636562	—	0.020711	0.019766
141	316.306748	1760.538315	—	324.764556	1553.863509	—	581.364619	—	53492.903899	—	—	—
142	19087.74132	7617.216223	—	7985.135655	2816.079963	—	11329.124253	—	94.613364	2.35625	4.3742	4.3198
143	1366.040111	911.647681	276449.945679	1036.713112	289.520913	19940.278611	920.078497	—	322.674031	26.0385	25.6122	19.834
144	123.620257	48.776211	4578.112921	103.044823	49.222351	889.450646	50.936869	18719.814589	58.012759	7.69925	29.0828	25.2248
145	10220.163485	3579.304633	—	4518.959823	1892.739915	195178.361891	3591.859873	—	14.449028	—	0.020237	0.020199
146	431.186375	2229.712044	—	303.54938	2029.480115	—	790.264722	—	17189.157839	—	—	2680.2
147	2722.173031	3470.05217	—	1330.082958	2819.898872	79279.112411	3474.962189	—	1001.312386	36.4871	6482.91	35.3994
148	5722.641439	1428.404252	—	3392.129186	528.613186	—	2681.881427	—	3031.561392	131.535	—	14.4495
149	1718.389401	966.2694	—	867.117448	651.375984	29844.867283	977.570001	—	51.408694	0.243787	11.2596	10.9109
150	12234.922362	2556.248908	—	6021.087076	1168.666278	227890.684195	2555.77573	—	64.510353	1.5043	7.13668	6.99642
151	—	297111.445057	—	114382.469674	68331.242136	—	—	—	6710.435967	112.676	—	168.637
152	193418.650774	98753.904601	—	95226.26362	18840.126101	—	99796.94703	—	746.283156	16.5005	103.202	3.54168
153	18.500025	18.143742	18.230962	18.673249	18.729936	18.859515	20.583184	0.955184	24.226776	0.889272	2.02956	1.95862
154	16178.99724	5580.387414	—	6080.995557	1307.610773	218463.135251	5584.625148	—	894.196529	51.4926	1173.35	1075.82
155	1706.471367	262.884754	186709.296336	1400.151465	150.185959	38653.414697	266.434936	—	301.180884	15.8498	356.111	307.222
156	31033.207799	8394.946111	—	7682.692732	1677.868762	266766.920241	8391.430609	—	18.487265	0.25973	0.181208	0.135074
157	13632.066678	3316.652083	—	7622.875742	2817.598291	286465.070358	3303.005596	—	17.83602	0.489005	0.544121	0.513884
158	37299.559778	7427.001169	—	11932.253756	1323.329209	203194.289008	7477.675271	—	14.721081	0.078633	0.042891	0.037916
159	7171.197885	792.117961	—	3239.359109	334.305193	87721.851293	798.628823	—	12.219974	—	0.020812	0.019845
160	12624.955898	13079.569403	—	7537.781844	5965.265223	—	13112.392476	—	25.713391	4.11286	1.38871	1.35089
161	3014.453779	1052.694406	—	1732.460644	925.574537	54083.698377	1065.248821	86182.541929	137.968114	10.2965	65.2517	59.0592
162	321.953688	424.647958	5835.126812	313.305808	407.331353	2190.613647	430.417568	3684.230864	645.470005	246.814	5349.34	15.6676
163	69301.891827	20386.956949	—	35244.514003	3073.70916	—	20608.009451	—	1219.507269	185.466	—	103.31
164	50467.580967	12143.641613	—	22939.072069	2059.774596	—	12191.860051	—	491.701953	24.3906	344.826	295.121
165	900.119475	2527.189853	229450.949514	427.419542	1674.815145	14088.522186	1668.984261	—	669.73483	103.229	1855.68	19.1249
166	61877.566011	10592.16409	—	27231.728372	2829.030989	—	10634.998355	—	30.266735	0.886239	0.401529	0.404298
167	6025.840466	2106.595957	—	2014.666615	591.527781	180312.409365	2115.024527	—	15.29079	—	0.019568	0.019056
168	5578.988753	5999.51595	—	4688.487494	3021.671143	—	6068.657945	—	85924.730808	1912.41	—	1125.29
169	1964.589162	929.561866	—	1658.178446	832.435233	83643.695975	930.362201	—	619.524577	6.23493	132.517	98.2303
170	57352.819918	17147.205087	—	16842.37867	6171.83486	—	17241.939818	—	14.342761	—	0.021964	0.021871
171	6674.55902	3105.268986	—	5320.107196	1858.619998	—	3105.004937	—	1451.014447	319.929	475.491	442.283
172	460.425091	182.345901	28547.293964	331.240123	97.309232	11313.275309	184.7498	—	9.986372	—	0.021043	0.0201
173	156.454967	49.430627	1675.219344	119.312277	51.990281	455.264241	51.528292	121153.694297	31.821535	11.6271	12.1727	11.1268
174	1512.062476	210.260925	14771.283857	985.520936	149.916162	4233.676028	213.936386	89349.27674	161.733762	0.80294	181.652	153.046
175	35695.422827	9426.525207	—	15786.679614	998.035702	—	9502.333416	—	13.577139	—	0.020835	0.024717
176	2978.356668	3026.665628	—	2655.215613	1259.114731	—	3036.412232	—	11.557116	—	0.019768	0.022608
177	2218.149484	248.380404	80388.905348	1315.231019	82.5459	7762.349315	252.131091	31209.740359	25.54321	0.396494	0.829072	0.768782
178	4439.364074	4435.710826	—	1917.142861	3165.739478	130033.554527	4465.835882	—	17.594339	0.865652	0.556469	0.527812
179	1200.18496	545.444146	—	967.980932	444.301636	—	547.492509	—	26190.843044	301.023	—	1735.62
180	7118.959603	2853.85473	—	3151.209172	1550.101077	—	2858.229265	—	13.818348	—	0.026005	0.022246
181	21460.739773	12960.146468	—	9897.462238	2490.715842	—	13013.348313	—	477.228648	22.3538	53.5208	47.2678
182	4918.740025	576.467824	—	2593.598973	336.821784	116470.332433	581.837212	—	673.28537	189.727	—	287.116
183	26654.568844	8278.120658	—	15026.973824	2336.326819	—	8327.205193	—	663.222007	26.2672	293.646	280.087
184	105082.351867	9318.458338	—	37719.803386	2182.231576	—	12035.521509	—	319.625418	3.10239	98.2574	85.5925
185	1841.308396	1107.083979	—	1351.742624	843.160771	21917.655036	1115.405179	—	55.302201	0.900347	10.5427	10.4077
186	5400.538605	10210.82647	—	2320.610713	7573.119709	—	10132.022664	—	29611.582465	—	—	1141.17
187	77.348688	33.063103	319.82038	75.964589	31.691205	328.309881	35.770088	597.799583	8.382514	—	0.023103	0.020895
188	52705.399489	12313.960524	—	21620.465995	11067.986696	—	12369.005884	—	19.019091	0.118786	0.047728	0.043407
189	2166.855752	667.440482	—	1863.451564	631.06023	51599.980016	668.998207	—	136.07284	2.33931	61.6339	44.8029
190	3423.156429	1180.704461	—	1983.868365	580.904182	104803.961273	1216.911949	—	261.294819	17.1509	124.224	97.7855
191	37006.790973	43987.657949	—	16835.994323	5262.302384	—	44211.67202	—	13.15319	—	0.02037	0.02033
192	3120.547025	406.631921	—	2021.676032	181.53723	96330.613703	409.170327	—	56.432046	0.911682	5.34221	5.0989
193	7612.3916	3345.973773	—	4015.510995	2280.830814	251820.036698	3365.77402	—	12.196418	—	0.023429	0.021565
194	9650.589142	1239.938393	—	3597.52765	271.520215	—	1247.186943	—	40.559617	4.81336	3.06805	2.72841
195	4594.91338	2552.50645	—	3396.514908	681.968567	—	2555.767786	—	6782.573859	—	—	140.145
196	137241.710162	12255.087332	—	43988.323536	11062.177676	—	12455.915282	—	19.57524	0.306246	0.325604	0.264715
197	266388.935722	6683.160582	—	79972.106672	2508.379313	—	6582.474163	—	11.538611	—	0.020827	0.020185
198	799.534564	1001.008714	16531.158981	412.304098	255.222232	1187.619676	998.926169	—	12.746187	0.08157	0.110509	0.116022
199	4591.594191	745.938874	—	2847.790609	433.639815	207420.804878	748.746559	—	791.821771	122.295	431.142	391.303
200	134.885049	197.101716	10687.9206	129.829116	184.091086	6110.367231	200.996485	—	159.586774	32.8202	154.981	132.285

Table 26: Results on *Single Counter*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
counter_01	18.058177	18.350703	18.060843	18.169518	18.933748	18.692677	19.607023	16.453027	6.978378	0.2575	0.118257	0.112888
counter_02	24.204002	31.358584	23.964412	47.131896	50.575059	47.358771	32.82203	211.518911	8.89128	2.61965	0.280484	0.288077
counter_03	67.810516	114.23335	104.986782	134.0321	171.003919	150.686689	119.903069	693.556915	12.781949	90.6461	16.9257	0.889436
counter_04	428.059632	831.486601	698.842123	588.71692	944.899279	944.688069	787.3057	2522.781763	25.034847	1218.46	27.5305	3.34075
counter_05	3754.291805	7568.91302	6852.560534	4065.36963	7778.35135	5770.683935	6975.687144	—	67.723467	13715.5	58.4356	381.417
counter_06	34590.517551	70189.903347	51550.536626	35831.825415	70901.656923	48650.17893	65032.73378	—	248.148853	143515.0	484.092	3020.43
counter_07	—	—	—	—	—	—	—	—	1107.050207	—	4089.62	27384.9
counter_08	—	—	—	—	—	—	—	—	5562.659418	—	30637.1	—
counter_09	—	—	—	—	—	—	—	—	27463.526719	—	—	—
counter_10	—	—	—	—	—	—	—	—	125102.602996	—	—	—
counter_11	—	—	—	—	—	—	—	—	—	—	—	—

Table 27: Results on *Uright*, time in milliseconds.

name	Nike Hash True First	Nike Hash False First	Nike Hash Random	Nike Bdd True First	Nike Bdd False First	Nike Bdd Random	Nike Multithreaded	Cynthia	Lydia	Lisa Symbolic	Lisa Explicit	Lisa
uright01	14.999567	15.462853	14.883559	14.865238	14.998477	13.940153	17.356649	1.995103	5.247303	0.138404	0.200602	0.055955
uright02	15.094761	15.38919	15.500275	14.319042	14.88239	14.062963	17.449376	0.702015	5.354601	0.064368	0.063398	0.068017
uright03	16.635718	15.695066	15.027712	14.483598	15.092849	14.438113	17.078239	0.620635	5.450475	0.057319	0.058338	0.059256
uright04	16.45286	16.809773	15.015942	14.84263	14.561528	15.816918	17.335316	0.715294	5.706349	0.046205	0.041596	0.040328
uright05	16.294455	16.307927	15.276146	15.317786	14.232395	14.600184	17.449579	0.30296	6.080545	0.042322	0.041693	0.042551
uright06	16.366488	15.497276	15.429629	15.02954	14.025922	14.222141	17.509895	0.695206	6.644925	0.065597	0.047732	0.048302
uright07	16.42893	16.002619	15.165841	15.164762	13.981411	14.432798	17.490425	0.692417	7.502385	0.052625	0.052897	0.063675
uright08	16.378655	16.174357	15.041196	15.793256	14.303577	14.115701	17.550876	0.579435	9.404088	0.093533	0.087099	0.089628
uright09	16.076093	16.063872	15.394778	15.188866	14.070482	14.398683	17.197007	0.557215	12.554514	0.139596	0.153389	0.136166
uright10	16.442906	15.984441	15.796126	14.347479	14.307109	14.676375	17.338181	0.564401	18.952341	0.194126	0.195148	0.196753
uright11	16.356052	15.645867	15.502792	14.265181	14.434716	14.739665	17.456657	0.490677	53.599676	0.303638	0.309067	0.334009
uright12	16.20536	16.260699	15.948424	14.291558	14.712576	15.052945	17.538113	0.990094	135.791584	0.486728	0.479552	0.479855
uright13	16.347945	15.890228	15.679354	14.140014	15.432763	15.739051	17.516188	0.43267	374.838683	0.861209	0.909934	0.87649
uright14	16.184014	15.617293	16.58412	14.618667	14.841335	15.104607	17.572259	0.72968	982.782789	1.59976	1.59355	9.61871
uright15	16.557987	15.913101	15.912645	14.744371	14.667065	14.96852	17.549767	0.490422	2566.476664	27.3199	15.2491	19.1711
uright16	16.39883	16.23711	16.080889	15.02702	15.378931	14.773313	17.599218	0.557976	6131.468587	22.4675	22.4194	22.6145
uright17	16.846842	15.619689	16.134446	15.255978	15.36376	15.070621	17.666699	0.791338	18033.003794	37.0282	37.1113	37.1563
uright18	17.409017	15.844109	15.955344	14.858477	14.707132	15.251954	17.750981	0.551119	—	81.6887	83.0052	99.4015
uright19	16.963635	16.71334	16.158252	14.982198	14.950929	15.667326	17.848841	0.601733	—	—	—	—
uright20	17.155057	16.049239	16.167525	15.394496	14.773912	16.415535	18.05109	1.001365	—	—	—	—