Univ rsité Paris 7 - L3 Informatiqu Anné 2008–2009

Introduction aux machines virtuelles

Examen du 19 mai 2009

Duré : 3h. Docum nts autorisés

I) Caml (∼8 points)

Exercice 1 Traduir n byt cod Caml l s phras s suivant s :

- `10-8*4/(3+1)`
- `let c rre x = x*x in c rre 5`
- `let = 10 in let decuple x = * x in decuple 5`
- `let f n = if n = 0 then 0 else n + f (n-1) in f 3`
- `let t = [|5;8|] in t.(0) + t.(1)`
  Donn r d ux manièr s d'accéd r aux cas s du tabl au `t`. Laqu ll d c s d ux méthod s r st utilisabl si on accèd à la cas `n` pour `n` non connu à l'avanc ?

Exercice 2 Étant donné la portion d byt cod Caml suivant :

```
        closurerec 1, 0
        const [1: [1: [0: 10] [0: 20]] [0: 30]]
        push
         cc 1
         pply 1
        return 2
L1:      cc 0
        switch/ 3 2
L3:      cc 0
        getfield 0
        return 1
L2:      cc 0
        getfield 1
        push
        offsetclosure 0
         pply 1
        push
         cc 1
        getfield 0
        push
        offsetclosure 0
         pply 1
         ddint
        return 1
```

On rapp ll qu la syntax `[n: b ...]` corr spond à la création d'un bloc d tag `n` t d donné s , `b`, tc.

- Propos r un typ caml corr spondant à la donné n cours d manipulation. R trouv r un phras Caml dont la compilation produit c byt cod . Associ r à chaqu élém nt d c tt phras la parti byt cod qui corr spond. Qu l st l résultat d l' xécution ?
- On xécut c byt cod jusqu'au pr mi r passag par l'instruction `offsetclosure`. Décrir alors l'état d la machin virtu ll Caml après c tt instruction : cont nu d l'accumulat ur, d la pil t du tas. Justifi r brièv m nt.
- Par qu ll instruction équival nt mais plus fficac p ut-on r mplac r l s instructions 5 t 6 (`pply 1` t `return 2`) ? Qu l st l' ff t d c chang m nt sur l'état d la pil d mandé à la qu stion précéd nt ?

II) Ja a (∼8 points)

Exercice 3 Traduir n byt cod Java l s instructions suivant s (n sachant qu `x` t `y` sont l s variabl s local s d'indic 0 t 1).

- `y=10-8*4/(3+x);`
- `y=(3+x)*4/8-10;`
- `y=1; for (x=1; x<3; x++) y=y*x;`
- `int [] y = new int [2]; x = y[0]+y[1];`

Dans chacun d s cas, simul r l'évolution d la pil lors d l' xécution (on suppos à chaqu fois qu `x` st nul initial m nt) t donn r la taill d pil néc ssair pour l bon déroul m nt du calcul.

Exercice 4 Étant donné la portion d byt cod Java qui suit :

- Propos r un portion d cod Java dont la compilation produit c byt cod . Associ r à chaqu élém nt d cod Java la parti byt cod qui corr spond.
- Qu l g nr d donné s n ntré p ut m n r c programm à un rr ur. Ajout r d s t sts p rm ttant d' mpêch r c la.
- Propos r un variant équival nt au byt cod précéd nt, mais où l `goto` s situ ntr d ux lab ls t non avant.

```
.method public st tic histo([I)[I
      .limit loc ls 4
      .limit st ck 4
      bipush 21
      new rr y int
       store_1
      iconst_0
      istore_2
      goto L2
L1:
       lo d_0
      ilo d_2
      i lo d
      istore_3
       lo d_1
      ilo d_3
       lo d_1
      ilo d_3
      i lo d
      iconst_1
      i dd
      i store
      iinc 2 1
L2:
      ilo d_2
       lo d_0
       rr ylength
      if_icmplt L1
       lo d_1
       return
.end method
```

III) Problème : alidation de bytecode (∼ 6 points)

Par défaut, un JVM comm nc par vérifi r qu l byt cod Java qu' ll doit xécut r satisfait bi n un c rtain nombr d critèr s d validité. En particuli r, ll ff ctu un contrôl du nombr d variabl s local s t d cas s d pil néc ssair à l' xécution du cod . On ch rch ici à voir comm nt c contrôl p ut êtr réalisé.

On suppos qu l byt cod Java nous st donné sous form d'un typ abstrait Caml :

```
type instruction =
 | Iconst of int
 | Istore of int
 | Ilo d of int
 | I dd
 | ...

type bytecode = instruction list
```

Exercice 5 Ecrir un fonction dét rminant l'indic maximal d s variabl s local s utilisé s dans un portion d byt cod . P ut-il arriv r qu l' xécution d c byt cod utilis n pratiqu moins d variabl local s qu c la ? Plus ?

Exercice 6 Qu ls problèm s s pos nt lorsqu l'on t nt d dét rmin r la quantité d pil maximal néc ssair pour xécut r un portion d byt cod ? C problèm st-il solubl n tout généralité ? Décrir comm nt calcul r c tt taill maximal pour l cas d s instructions n' ntraînant pas d sauts.

Au li u d s véritabl s instructions d saut du byt cod Java, on suppos maint nant dispos r d'instructions d plus haut niv au :

```
type instruction =
  ...
  | If_icmplt of bytecode * bytecode   (* code du then / du else *)
  | While of bytecode * bytecode * bytecode
            (* code d'initi lis tion / du test de fin / du corps de l  boucle *)
  ...

 nd bytecode = instruction list
```

Exercice 7 Dans c cadr , qu ll contraint doit satisfair l byt cod s trouvant dans un `if` ou un `while` pour qu l'on soit n m sur d dét rmin r un taill d pil maximal à l'avanc ? Ecrir c tt fonction d calcul d la taill d pil maximal .

Exercice 8 On souhait égal m nt vérifi r qu chaqu instruction trouv ra bi n un quantité d pil suffisant d vant ll lors d l' xécution. Ainsi un `i dd` néc ssit l'accès à d ux cas s d pil . Qu faut-il chang r aux qu stions précéd nt s pour êtr n m sur d t st r c tt propriété ?

Exercice 9 C s t chniqu s d'analys d taill d pil s p uv nt- ll s s'appliqu r égal m nt dans l cas du byt cod Caml ?