

Organisation des ordinateurs
Examen de seconde session 2023
Énoncés et solutions

Énoncés

1. Un accéléromètre mesure des accélérations comprises entre -30 et $+30$ m/s^2 (ces valeurs étant incluses). La résolution de ce capteur diffère selon la plage de mesure : entre $+5$ et $+15$ m/s^2 , les mesures comprennent un chiffre décimal après la virgule (par exemple : $+9,8$ m/s^2). En dehors de cet intervalle, les mesures sont entières (par exemple : -9 m/s^2). On ne distingue pas -0 m/s^2 de $+0$ m/s^2 .
 - (a) Si toutes les mesures que peut produire l'accéléromètre sont équiprobables, quelle quantité d'information une telle mesure contient-elle ?
 - (b) On utilise un canal de transmission d'une capacité de 100 Mb/s pour communiquer les mesures fournies par l'accéléromètre à un ordinateur. Si ce canal est utilisé le plus efficacement possible combien de temps faudra-t-il pour transmettre 10^9 mesures ?

2.
 - (a) Pour une suite de n bits w , on note respectivement $[w]_{ns}$ et $[w]_{c2}$ les nombres entiers représentés par w selon les représentations non signée et par complément à deux. On demande d'exprimer $[w]_{c2}$ en fonction de $[w]_{ns}$ et de n .
 - (b) Calculer la somme $(-2) + (-2)$ en représentant les nombres par complément à un sur 6 bits. (Détailler toutes les étapes du calcul.)
 - (c) Calculer le produit $\frac{3}{2} \times 5$ en virgule fixe. Vous êtes libre de choisir la taille des représentations des opérandes et du résultat.
 - (d) Construire la représentation en double précision du plus grand nombre réel représentable de façon exacte selon le standard IEEE754, et en donner une écriture hexadécimale.

3.
 - (a) À quoi les drapeaux d'un processeur servent-ils ? Dans quel composant du circuit du processeur sont-ils mémorisés ?
 - (b) Expliquer le principe de fonctionnement de la pile pour l'architecture x86-64. En particulier, que contient le registre RSP ?

- (c) Expliquer étape par étape comment se déroulera l'exécution du fragment de code assembleur x86-64 suivant, en faisant l'hypothèse que la pile est initialement correctement configurée. Quelle sera la valeur finale de `RAX` ?

```
MOV  RAX, -1
XOR  RAX, 8
NOT  RAX
SUB  RSP, RAX
MOV  qword ptr[RSP], RAX
MOV  byte ptr[RSP + 1], AL
POP  RAX
```

4. On souhaite programmer une fonction `compter_seuil` acceptant en arguments un tableau d'entiers `t`, le nombre d'éléments `n` de celui-ci, et un entier `v`. Cette fonction doit retourner le nombre d'éléments de `t` qui sont supérieurs ou égaux à `v`. Les éléments de `t` et le paramètre `v` sont représentés de façon signée sur 32 bits.
- (a) Écrire, en pseudocode ou en langage C (au choix), un algorithme permettant de résoudre ce problème.
- (b) Traduire cet algorithme en un programme assembleur x86-64, en veillant à respecter la convention d'appel de fonctions des systèmes *Unix*.

Exemples de solutions

1. (a) L'intervalle $[-30, 30]$ contient 61 valeurs entières. Il y a en outre 9 valeurs décimales supplémentaires à considérer dans chaque intervalle $[5,1, 5,9]$, $[6,1, 6,9]$, \dots , $[14,1, 14,9]$. On a donc au total $61 + 10 \times 9 = 151$ valeurs distinctes pouvant être produites par l'accéléromètre. Étant donné que ces valeurs sont équiprobables, la quantité d'information que chacune d'entre elles contient vaut

$$\log_2 151 \approx 7,238 \text{ bits.}$$

- (b) La quantité d'information correspondant à 10^9 mesures vaut $7,238 \cdot 10^9$ bits. À la vitesse de $100 \cdot 2^{20}$ bits/s, il faudra donc

$$\frac{7,238 \cdot 10^9}{100 \cdot 2^{20}} \approx 69 \text{ s}$$

pour transmettre ces données.

3. (a) Les drapeaux d'un processeur fournissent des informations sur la façon dont l'exécution de certaines instructions s'est déroulée, en indiquant par exemple si des opérations arithmétiques ont produit un report, un dépassement ou un résultat nul. Ils sont mis à jour par l'unité arithmétique et logique (ALU), et mémorisés dans la banque de registres du processeur.
- (b) Dans l'architecture x86-64, la pile occupe une zone de la mémoire qui croît dans la direction des adresses décroissantes, et le registre `RSP` contient toujours l'adresse du dernier octet qui a été empilé (c'est-à-dire celui situé au sommet de la pile). Une opération d'empilement revient donc à diminuer `RSP`, et à écrire la valeur empilée en mémoire à l'endroit pointé par la nouvelle valeur de `RSP`. Réciproquement, une valeur est dépilée en la recopiant depuis l'endroit pointé par `RSP`, et en augmentant ensuite la valeur de ce registre.
- (c) La première instruction met à 1 tous les bits du registre `RAX`. Ensuite, l'instruction `XOR` complémente le bit situé à la position 3, et l'instruction `NOT` complémente ensuite tous les bits de ce registre. À ce stade, seul le bit à la position 3 de `RAX` est donc égal à 1, ce qui signifie que `RAX` contient la valeur 8. L'instruction `SUB` décrémente donc `RSP` de 8 unités, et l'instruction suivante empile la valeur courante de `RAX`, c'est-à-dire l'entier 8 représenté sur 64 bits. L'avant-dernière instruction modifie le deuxième octet de cette valeur présente sur la pile, en le remplaçant par l'octet de poids faible de `RAX`, qui vaut 8. La dernière instruction placera donc dans `RAX` l'entier `0x808` extrait de la pile.
4. (a) `unsigned long compter_seuil(int t[], unsigned long n, int v)`
- ```

{
 unsigned long i, nb;

 nb = 0;

 for (i = 0; i < n; i++)
 if (t[i] >= v)
 nb++;

 return nb;
}

```

```
(b) .intel_syntax noprefix
 .text
 .global compter_seuil
 .type compter_seuil, @function
compter_seuil:
 PUSH RBP
 MOV RBP, RSP
 MOV RAX, 0
 MOV RCX, 0
boucle: CMP RCX, RSI
 JAE fin
 CMP dword ptr [RDI + 4 * RCX], EDX
 JL suite
 INC RAX
suite: INC RCX
 JMP boucle
fin: POP RBP
 RET
```