

INF2171 — Organisation des ordinateurs et assembleur

Examen intra hiver 2024

Jean Privat & Hugo Leblanc — Dimanche 25 février

Durée 3 heures

- *Aucun document n'est autorisé sauf une feuille de note manuscrite au format lettre (recto-verso).*
- *L'usage de la calculatrice ou de tout appareil électronique est interdit.*
- *Répondez et rendez seulement le formulaire (avant dernière feuille).*
- *L'annexe est détachable (dernière feuille).*
- *Les questions faisant apparaître le symbole ♣ peuvent présenter plusieurs bonnes réponses. Les autres ont une unique bonne réponse.*
- *Chacune des 20 questions vaut 5 points. Des points négatifs pourront être affectés à de très mauvaises réponses.*

1 Structure et fonctionnement d'un ordinateur

Question 1 Quelle est l'alternative classique à l'architecture de type RISC ?

- | | |
|--|---|
| <input type="checkbox"/> A Engine Instruction Set Computer | <input type="checkbox"/> F Hard Instruction Set Computer |
| <input type="checkbox"/> B Spaced Instruction Set Computer | <input type="checkbox"/> G Dedicated Instruction Set Computer |
| <input type="checkbox"/> C General Instruction Set Computer | <input type="checkbox"/> H Elaborate Instruction Set Computer |
| <input checked="" type="checkbox"/> D Complex Instruction Set Computer | <input type="checkbox"/> I Friendly Instruction Set Computer |
| <input type="checkbox"/> E Compiled Instruction Set Computer | <input type="checkbox"/> J eXtrem Instruction Set Computer |

2 Arithmétique des ordinateurs

Question 2 ♣ Parmi les représentations suivantes, quelles sont celles du nombre $33_{(16)}$ (33 hexadécimal) ? Indiquez toutes les bonnes réponses.

- | | | | |
|---|--|---|--|
| <input type="checkbox"/> A $52_{(10)}$ | <input checked="" type="checkbox"/> C $110011_{(2)}$ | <input type="checkbox"/> E $41_{(8)}$ | <input type="checkbox"/> G $41_{(16)}$ |
| <input type="checkbox"/> B $011011_{(2)}$ | <input type="checkbox"/> D $21_{(16)}$ | <input checked="" type="checkbox"/> F $51_{(10)}$ | <input checked="" type="checkbox"/> H $63_{(8)}$ |

Question 3 ♣ Parmi les séquences d'instructions RISC-V suivantes, quelles sont celles qui positionnent le registre $s0$ à la valeur $0x123$? Indiquez toutes les bonnes réponses.

- | | | |
|--|---|---|
| <input type="checkbox"/> A <code>li s0, 0x100</code>
<code>addi s0, s0, 23</code> | <input checked="" type="checkbox"/> C <code>li s0, 0x1FE</code>
<code>xori s0, s0, 0xDD</code> | <input checked="" type="checkbox"/> E <code>li s0, 0x1230</code>
<code>srli s0, s0, 4</code> |
| <input checked="" type="checkbox"/> B <code>li s0, 0x146</code>
<code>addi s0, s0, -35</code> | <input type="checkbox"/> D <code>li s0, 0x678</code>
<code>andi s0, s0, 0x123</code> | <input type="checkbox"/> F <code>li s0, 0x1230</code>
<code>slli s0, s0, 1</code> |

Question 4 ♣ Quelles opérations causent un débordement en considérant la taille maximale des résultats ? Indiquez toutes les bonnes réponses.

- | | |
|---|---|
| <input checked="" type="checkbox"/> A Résultat sur 8 bits signé, $70 + 70$ | <input type="checkbox"/> E Résultat sur 16 bits signé, $-15000 - 15000$ |
| <input type="checkbox"/> B Résultat sur 8 bits non signé, $70 + 70$ | <input type="checkbox"/> F Résultat sur 8 bits signé, $-1 - -1$ |
| <input checked="" type="checkbox"/> C Résultat sur 8 bits signé, $-1 + -128$ | <input checked="" type="checkbox"/> G Résultat sur 16 bits non-signé, $0x1234 * 0x10$ |
| <input checked="" type="checkbox"/> D Résultat sur 16 bits non-signé, $0xFEDC + 0xBA98$ | <input type="checkbox"/> H Résultat sur 32 bits non-signé, $0x1234 * 0x5678$ |

brouillon

3 Instructions et code machine RISC-V

Question 5 Que fait la directive RISC-V « `.word 0x10` » ?

- A Réserve 2 octets, le premier initialisé à 0 et le dernier initialisé à 0x10 (hexa)
- B Réserve 10 octets, le dernier initialisé à 0x4 (hexa) et les autres à 0
- C Réserve 16 octets, le premier initialisé à 0x10 (hexa) et les autres à 0
- D Réserve 4 octets, le dernier initialisé à 0x10 (hexa) et les autres à 0
- E Réserve 2 octets, le premier initialisé à 0x10 (hexa) et le second à 0
- F Réserve 16 octets, le dernier initialisé à 0x4 (hexa) et les autres à 0
- G Réserve 10 octets, tous initialisé à 0x10 (hexa)
- H Réserve 16 octets, tous initialisé à 0x10 (hexa)
- I Réserve 4 octets, le premier initialisé à 0x10 (hexa) et les autres à 0
- J Réserve un seul octet, initialisé à 0x10 (hexa)
- K Réserve 10 octets, le dernier initialisé à 0x10 (hexa) et les autres à 0
- L Réserve 16 octets, tous initialisé à 0x04 (hexa)
- M Réserve 10 octets, le premier initialisé à 0x4 (hexa) et les autres à 0
- N Réserve 10 octets, le premier initialisé à 0x10 (hexa) et les autres à 0
- O Réserve 10 octets, tous initialisé à 0x04 (hexa)
- P Réserve 16 octets, le dernier initialisé à 0x10 (hexa) et les autres à 0
- Q Réserve 16 octets, le premier initialisé à 0x4 (hexa) et les autres à 0

Question 6 Désassemblez l'instruction machine RISC-V 64 bits qui est codée en mémoire par les 4 octets suivants : « B3 C4 65 02 ».

- | | | | |
|---|---|---|--|
| <input type="checkbox"/> A <code>div t1, a1, s1</code> | <input type="checkbox"/> D <code>xori a1, t1, 41</code> | <input type="checkbox"/> G <code>xori s1, a1, 38</code> | <input type="checkbox"/> J <code>xori s1, a1, 6</code> |
| <input type="checkbox"/> B <code>xori t1, a1, 41</code> | <input type="checkbox"/> E <code>xor s1, a1, t1</code> | <input type="checkbox"/> H <code>div s1, a1, t1</code> | <input type="checkbox"/> K <code>xori t1, a1, 9</code> |
| <input type="checkbox"/> C <code>divi s1, a1, 6</code> | <input type="checkbox"/> F <code>xor t1, a1, s1</code> | <input type="checkbox"/> I <code>xori a1, s1, 38</code> | <input type="checkbox"/> L <code>divi t1, a1, 9</code> |

Question 7 ♣ Dans une architecture RISC-V 64 bits, quelles sont les directives qui génèrent des données d'exactly la même taille qu'un registre. Indiquez toutes les bonnes réponses.

- | | | |
|--|---|--|
| <input type="checkbox"/> A <code>.byte 1, 2, 3, 4</code> | <input type="checkbox"/> F <code>.word 0,0,0,0</code> | <input type="checkbox"/> K <code>.align 2</code> |
| <input type="checkbox"/> B <code>.space 8</code> | <input type="checkbox"/> G <code>.half 0, 1, 2, 3</code> | <input type="checkbox"/> L <code>.eqv size, 8</code> |
| <input type="checkbox"/> C <code>.eqv size, 64</code> | <input type="checkbox"/> H <code>.half 0xFF, 0xFF</code> | <input type="checkbox"/> M <code>.ascii "64"</code> |
| <input type="checkbox"/> D <code>.dword 64</code> | <input type="checkbox"/> I <code>.string "bonbons"</code> | <input type="checkbox"/> N <code>.dword 0</code> |
| <input type="checkbox"/> E <code>.word 2</code> | <input type="checkbox"/> J <code>.space 4</code> | <input type="checkbox"/> O <code>.space 64</code> |

brouillon

4 Programme mystère

Soit le programme mystère suivant :

```

    li a7, 5
    ecall
    mv s0, a0
    ecall
    mv s1, a0
    li s2, 0
aaaa:
    beqz s0, bbbb
    add s2, s2, s1
    addi s0, s0, -1
    j aaaa
bbbb:
    li a7, 1
    mv a0, s2
    ecall
    li a7, 10
    ecall

```

Question 8 Qu'affiche-t-il lorsque l'on saisit les nombres 0 et 10 ?

- A 0
- B 1
- C 2
- D 3

- E 4
- F 5
- G 6
- H 7

- I 8
- J Il n'affiche rien du tout
- K Il part en boucle infinie
- L Il affiche n'importe quoi

Question 9 Qu'affiche-t-il lorsque l'on saisit les nombres 2 et 3 ?

- A 0
- B 1
- C 2
- D 3

- E 4
- F 5
- G 6
- H 7

- I 8
- J Il n'affiche rien du tout
- K Il part en boucle infinie
- L Il affiche n'importe quoi

Question 10 Quel est le premier caractère affiché lorsque l'on saisit les nombres 2000 et 4000 ?

- A 0
- B 1
- C 2
- D 3

- E 4
- F 5
- G 6
- H 7

- I 8
- J Il n'affiche rien du tout
- K Il part en boucle infinie
- L Il affiche n'importe quoi

brouillon

5 Modes d'adressage

Soit le morceau de programme suivant exécuté sur une architecture RISC-V 64 bits. Sachant que l'étiquette `cafe` est associée à l'adresse `0x10010000`, indiquez quel est le contenu (hexadécimal) final des registres demandés.

```
.data
cafe:
    .word 0xCAFE
    .eqv  CA, 0xFE
    .byte  CA, 0xFE
    .ascii "CAFE"

.text
la s0, cafe
lw s1, cafe
li s2, CA
lb s3, 1(s0)
lwu s4, 4(s0)
```

Question 11 Le registre `s0`.

- | | | | | | | | |
|----------------------------|--------------------|----------------------------|--------------------|----------------------------|--------------------|---------------------------------------|--------------------|
| <input type="checkbox"/> A | 0x0000100100000000 | <input type="checkbox"/> E | 0x0800100100000000 | <input type="checkbox"/> I | 0x0000000001100000 | <input type="checkbox"/> M | 0x0000000001100080 |
| <input type="checkbox"/> B | 0x0000011000000000 | <input type="checkbox"/> F | 0x0110008000000000 | <input type="checkbox"/> J | 0x1001000000000000 | <input type="checkbox"/> N | 0x0000000000000110 |
| <input type="checkbox"/> C | 0x0110000000000000 | <input type="checkbox"/> G | 0x0000000008001001 | <input type="checkbox"/> K | 0x0000000000001001 | <input checked="" type="checkbox"/> O | 0x0000000010010000 |
| <input type="checkbox"/> D | 0x0000000008000110 | <input type="checkbox"/> H | 0x0000000010010008 | <input type="checkbox"/> L | 0x8000011000000000 | <input type="checkbox"/> P | 0x1001000800000000 |

Question 12 Le registre `s1`.

- | | | | | | | | |
|----------------------------|--------------------|----------------------------|--------------------|---------------------------------------|--------------------|----------------------------|--------------------|
| <input type="checkbox"/> A | 0xCAFE000000000000 | <input type="checkbox"/> E | 0x000000000000FEFE | <input type="checkbox"/> I | 0x000000000000CACA | <input type="checkbox"/> M | 0x0000000043414645 |
| <input type="checkbox"/> B | 0x0000000045464143 | <input type="checkbox"/> F | 0x00000000CAFEFEFE | <input type="checkbox"/> J | 0x000000000000FECA | <input type="checkbox"/> N | 0xCAFEFEFE00000000 |
| <input type="checkbox"/> C | 0x00000000000000FE | <input type="checkbox"/> G | 0x00000000000000CA | <input type="checkbox"/> K | 0xFE00000000000000 | <input type="checkbox"/> O | 0xFECA000000000000 |
| <input type="checkbox"/> D | 0x4341464500000000 | <input type="checkbox"/> H | 0xCA00000000000000 | <input checked="" type="checkbox"/> L | 0x000000000000CAFE | <input type="checkbox"/> P | 0x4546414300000000 |

Question 13 Le registre `s2`.

- | | | | | | | | |
|---------------------------------------|--------------------|----------------------------|--------------------|----------------------------|--------------------|----------------------------|--------------------|
| <input type="checkbox"/> A | 0xFE00000000000000 | <input type="checkbox"/> E | 0xCAFE000000000000 | <input type="checkbox"/> I | 0x000000000000CACA | <input type="checkbox"/> M | 0x0000000045464143 |
| <input type="checkbox"/> B | 0x00000000CAFEFEFE | <input type="checkbox"/> F | 0xCA00000000000000 | <input type="checkbox"/> J | 0x00000000000000CA | <input type="checkbox"/> N | 0x000000000000FECA |
| <input checked="" type="checkbox"/> C | 0x00000000000000FE | <input type="checkbox"/> G | 0xCAFEFEFE00000000 | <input type="checkbox"/> K | 0xFECA000000000000 | <input type="checkbox"/> O | 0x4341464500000000 |
| <input type="checkbox"/> D | 0x4546414300000000 | <input type="checkbox"/> H | 0x000000000000FEFE | <input type="checkbox"/> L | 0x000000000000CAFE | <input type="checkbox"/> P | 0x0000000043414645 |

Question 14 Le registre `s3`.

- | | | | | | | | |
|----------------------------|--------------------|----------------------------|--------------------|---------------------------------------|----------------------|----------------------------|----------------------|
| <input type="checkbox"/> A | 0x00000000CAFEFEFE | <input type="checkbox"/> E | 0xCAFEFFFFFFFFFFFF | <input type="checkbox"/> I | 0xFFFFFFFFFFFFFFFFFE | <input type="checkbox"/> M | 0x00000000000000FE |
| <input type="checkbox"/> B | 0xFEFEFFFFFFFFFFFF | <input type="checkbox"/> F | 0x0000000000000000 | <input type="checkbox"/> J | 0xFEFFFFFFFFFFFFFFFF | <input type="checkbox"/> N | 0x0000000000FEFEFE |
| <input type="checkbox"/> C | 0x0000000000000043 | <input type="checkbox"/> G | 0xFEFE000000000000 | <input type="checkbox"/> K | 0xFFFFFFFFFFFEFEFE | <input type="checkbox"/> O | 0xFFFFFFFFFFFFFFFFFE |
| <input type="checkbox"/> D | 0x000000000000FEFE | <input type="checkbox"/> H | 0x00000000000000CA | <input checked="" type="checkbox"/> L | 0xFFFFFFFFFFFFFCA | <input type="checkbox"/> P | 0xFFFFFFFFCAFEFEFE |

Question 15 Le registre `s4`.

- | | | | | | | | |
|----------------------------|--------------------|----------------------------|----------------------|----------------------------|--------------------|---------------------------------------|--------------------|
| <input type="checkbox"/> A | 0x00000000FEFE4143 | <input type="checkbox"/> E | 0x00000000CAFECAFE | <input type="checkbox"/> I | 0x00000000FE434146 | <input type="checkbox"/> M | 0x000000000000FECA |
| <input type="checkbox"/> B | 0x4341FE4143FE4143 | <input type="checkbox"/> F | 0xFFFFFFFFFFFFFFFFFE | <input type="checkbox"/> J | 0x00000000CAFECAFE | <input type="checkbox"/> N | 0x00000000CAFE4341 |
| <input type="checkbox"/> C | 0x4143FE4341464500 | <input type="checkbox"/> G | 0x0000000000004645 | <input type="checkbox"/> K | 0x0000000043414645 | <input type="checkbox"/> O | 0x00000000000000FE |
| <input type="checkbox"/> D | 0x000000000000CAFE | <input type="checkbox"/> H | 0x00000000CAFEFEFE | <input type="checkbox"/> L | 0x0000000045464143 | <input checked="" type="checkbox"/> P | 0x000000004143FEFE |

6 Programmation

Soit le programme incomplet suivant dont l'objectif est d'afficher la somme et le produit des éléments d'un tableau.

Note : pour simplifier l'exercice, le tableau est donné directement dans le programme (et non lu de l'entrée). Ainsi ce programme avec le tableau initial tel que donné devrait afficher « 33 336000 ». Si l'on remplaçait le premier mot de `tab` par 6 le programme afficherait « 38 2016000 ».

```
# Somme et produits de nombres dans un tableau
.data
# Tableau de 10 valeurs
tab: .word 1, 5, 8, -7, 2, 3, 10, 2, 10, -1
      .eqv tablen, 10 # Nombre d'éléments du tableau

# Appels système RARS utilisés
      .eqv PrintInt, 1
      .eqv PrintChar, 11
      .eqv Exit, 10

.text
# Initialisations # TODO1
la s0, tab      # s0 Adresse de l'élément courant
li s1, tablen   # s1 Nombre d'éléments restants à traiter
l?????        # s2 Somme actuelle
l?????        # s3 Produit actuel

loop:
# Boucler sur chaque élément # TODO2
b?????

# Récupérer dans s4 la valeur de l'élément courant # TODO3
l?????

# Mettre à jour la somme et le produit actuels # TODO4
a?????
m?????

# Passer à l'élément suivant # TODO5
a?????
a?????
j loop

end:
# Afficher somme et produit
li a7, PrintInt
mv a0, s2
ecall
li a7, PrintChar
li a0, ' '
ecall
li a7, PrintInt
mv a0, s3
ecall
li a7, Exit
ecall
```


Question 16 Que faut-il pour initialiser le registre s3 dans le groupe d'instructions TODO1 ?

- | | | | |
|---|--|--|--|
| <input type="checkbox"/> A lw s3, 0(tablen) | <input type="checkbox"/> D la s3, tab | <input type="checkbox"/> G lw s3, 0(tab) | <input type="checkbox"/> J lw s3, -1(loop) |
| <input type="checkbox"/> B lw s3, 1(loop) | <input checked="" type="checkbox"/> E li s3, 1 | <input type="checkbox"/> H li s3, tablen | <input type="checkbox"/> K li s3, -1 |
| <input type="checkbox"/> C li s3, tab | <input type="checkbox"/> F lw s3, -1(tab) | <input type="checkbox"/> I li s3, 0 | <input type="checkbox"/> L la s3, tablen |

Question 17 Que faut-il mettre à l'instruction manquante du TODO2 ?

- | | | | |
|---|--|--|---|
| <input type="checkbox"/> A bgt s1, s2, end | <input type="checkbox"/> E bgtz s1, loop | <input checked="" type="checkbox"/> I blez s1, end | <input type="checkbox"/> M ble s1, s2, loop |
| <input type="checkbox"/> B bgtz s1, end | <input type="checkbox"/> F bgez s2, loop | <input type="checkbox"/> J bgtz s2, loop | <input type="checkbox"/> N ble s1, s2, end |
| <input type="checkbox"/> C bgt s1, s2, loop | <input type="checkbox"/> G bgez s1, loop | <input type="checkbox"/> K blez s2, loop | <input type="checkbox"/> O bge s1, s2, loop |
| <input type="checkbox"/> D bgtz s2, end | <input type="checkbox"/> H bgez s2, end | <input type="checkbox"/> L blez s2, end | <input type="checkbox"/> P blez s1, loop |

Question 18 Que faut-il mettre à l'instruction manquante du TODO3 ?

- | | | | |
|---|---|---|--|
| <input type="checkbox"/> A ld s4, 0(s0) | <input type="checkbox"/> E lw s3, 0(s1) | <input type="checkbox"/> I ld s4, 4(s1) | <input checked="" type="checkbox"/> M lw s4, 0(s0) |
| <input type="checkbox"/> B ld s4, 0(s1) | <input type="checkbox"/> F ld s3, 4(s1) | <input type="checkbox"/> J ld s3, 0(s1) | <input type="checkbox"/> N ld s3, 0(s0) |
| <input type="checkbox"/> C lw s3, 4(s1) | <input type="checkbox"/> G lw s4, 0(s1) | <input type="checkbox"/> K lw s3, 0(s0) | <input type="checkbox"/> O lw s4, 4(s1) |
| <input type="checkbox"/> D lw s3, 4(s0) | <input type="checkbox"/> H ld s4, 4(s0) | <input type="checkbox"/> L lw s4, 4(s0) | <input type="checkbox"/> P ld s3, 4(s0) |

Question 19 Que faut-il mettre aux instructions manquantes du TODO4 ?

- | | | | |
|--|--|--|--|
| <input type="checkbox"/> A add s2, s4, s4 | <input type="checkbox"/> C addi s2, s4, s4 | <input type="checkbox"/> E addi s3, s3, s4 | <input type="checkbox"/> G addi s2, s2, s4 |
| <input type="checkbox"/> B mul s3, s4, s4 | <input type="checkbox"/> D muli s3, s4, s4 | <input type="checkbox"/> F muli s2, s2, s4 | <input type="checkbox"/> H muli s3, s3, s4 |
| <input type="checkbox"/> B addi s3, s4, s4 | <input type="checkbox"/> D add s3, s4, s4 | <input checked="" type="checkbox"/> F add s2, s2, s4 | <input type="checkbox"/> H add s3, s3, s4 |
| <input type="checkbox"/> B muli s2, s4, s4 | <input type="checkbox"/> D mul s2, s4, s4 | <input type="checkbox"/> F mul s3, s3, s4 | <input type="checkbox"/> H mul s2, s2, s4 |

Question 20 Que faut-il mettre aux instructions manquantes du TODO5 ?

- | | | | |
|--|--|--|--|
| <input type="checkbox"/> A addi s0, s0, 1 | <input type="checkbox"/> E addi s0, s0, -1 | <input checked="" type="checkbox"/> I addi s0, s0, 4 | <input type="checkbox"/> M addi s0, s0, 4 |
| <input type="checkbox"/> A addi s1, s1, 4 | <input type="checkbox"/> E addi s1, s1, -1 | <input type="checkbox"/> I addi s1, s1, -1 | <input type="checkbox"/> M addi s1, s1, -4 |
| <input type="checkbox"/> B addi s0, s0, -1 | <input type="checkbox"/> F addi s0, s0, 1 | <input type="checkbox"/> J addi s0, s0, 1 | <input type="checkbox"/> N addi s0, s0, 4 |
| <input type="checkbox"/> B addi s1, s1, -4 | <input type="checkbox"/> F addi s1, s1, -1 | <input type="checkbox"/> J addi s1, s1, -4 | <input type="checkbox"/> N addi s1, s1, 1 |
| <input type="checkbox"/> C addi s0, s0, -4 | <input type="checkbox"/> G addi s0, s0, -4 | <input type="checkbox"/> K addi s0, s0, -4 | <input type="checkbox"/> O addi s0, s0, 4 |
| <input type="checkbox"/> C addi s1, s1, 1 | <input type="checkbox"/> G addi s1, s1, -1 | <input type="checkbox"/> K addi s1, s1, 4 | <input type="checkbox"/> O addi s1, s1, 4 |
| <input type="checkbox"/> D addi s0, s0, -1 | <input type="checkbox"/> H addi s0, s0, -1 | <input type="checkbox"/> L addi s0, s0, 1 | <input type="checkbox"/> P addi s0, s0, -4 |
| <input type="checkbox"/> D addi s1, s1, 4 | <input type="checkbox"/> H addi s1, s1, 1 | <input type="checkbox"/> L addi s1, s1, 1 | <input type="checkbox"/> P addi s1, s1, -4 |

brouillon

brouillon



Feuille de réponses — Examen intra hiver 2024
INF2171 — Organisation des ordinateurs et assembleur
Jean Privat & Hugo Leblanc — Dimanche 25 février

0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9

← codez les 8 chiffres de votre code permanent ci-contre, et inscrivez-le à nouveau ci-dessous avec votre nom et prénom. 5 points de pénalité en cas d'oubli ou d'erreur.

Code permanent :

.....

Nom :

.....

Prénom :

.....

Les réponses aux questions sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte. Ne pas utiliser un formulaire d'une autre copie au risque d'avoir toutes les réponses fausses.

Important : noircissez complètement l'intérieur de chaque case (pas de croix, pas de cercle).

- Question 1 :

A	B	C	D	E	F	G	H	I	J
---	---	---	----------	---	---	---	---	---	---
- Question 2 :

A	B	C	D	E	F	G	H
---	---	----------	---	---	----------	---	----------
- Question 3 :

A	B	C	D	E	F
---	----------	----------	---	----------	---
- Question 4 :

A	B	C	D	E	F	G	H
----------	---	----------	---	---	---	----------	---
- Question 5 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
---	---	---	---	---	---	---	---	----------	---	---	---	---	---	---	---	---
- Question 6 :

A	B	C	D	E	F	G	H	I	J	K	L
---	---	---	---	---	---	---	----------	---	---	---	---
- Question 7 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
---	----------	----------	----------	---	---	----------	----------	----------	---	---	---	---	----------	---
- Question 8 :

A	B	C	D	E	F	G	H	I	J	K	L
----------	---	---	---	---	---	---	---	---	---	---	---
- Question 9 :

A	B	C	D	E	F	G	H	I	J	K	L
---	---	---	---	---	---	----------	---	---	---	---	---
- Question 10 :

A	B	C	D	E	F	G	H	I	J	K	L
---	---	---	---	---	---	---	---	----------	---	---	---
- Question 11 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	---	---	---	---	---	---	---	---	---	---	---	---	----------	---
- Question 12 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	---	---	---	---	---	---	---	---	---	----------	---	---	---	---
- Question 13 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	----------	---	---	---	---	---	---	---	---	---	---	---	---	---
- Question 14 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	---	---	---	---	---	---	---	---	---	----------	---	---	---	---
- Question 15 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	---	---	----------	---	---	---	---	---	---	---	---	---	---	----------
- Question 16 :

A	B	C	D	E	F	G	H	I	J	K	L
---	---	---	---	----------	---	---	---	---	---	---	---
- Question 17 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	---	---	---	---	---	---	----------	---	---	---	---	---	---	---
- Question 18 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	---	---	---	---	---	---	---	---	---	---	----------	---	---	---
- Question 19 :

A	B	C	D	E	F	G	H
---	---	---	---	---	----------	---	---
- Question 20 :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
---	---	---	---	---	---	---	---	----------	---	---	---	---	---	---	---

Jeu d'instruction RISC-V

Inst	Description	FMT	opcode	fn3	fn7 (ou imm)	Description	Note
add	Add	R	0x33	0x0	0x00	rd = rs1 + rs2	
sub	Substract	R	0x33	0x0	0x20	rd = rs1 - rs2	
xor	Exclusive Or	R	0x33	0x4	0x00	rd = rs1 ^ rs2	
or	Or	R	0x33	0x6	0x00	rd = rs1 rs2	
and	And	R	0x33	0x7	0x00	rd = rs1 & rs2	
sll	Shift Left Logical	R	0x33	0x1	0x00	rd = rs1 << rs2	
srl	Shift Right Logical	R	0x33	0x5	0x00	rd = rs1 >> rs2	
sra	Shift Right Arithmetic	R	0x33	0x5	0x20	rd = rs1 >> rs2	
slt	Set Less Than	R	0x33	0x2	0x00	rd = (rs1 < rs2)?1:0	
sltu	Set Less Than (U)	R	0x33	0x3	0x00	rd = (rs1 < rs2)?1:0	
addi	Add Immediate	I	0x13	0x0		rd = rs1 + imm	
xori	Xor Immediate	I	0x13	0x4		rd = rs1 ^ imm	
ori	Or Immediate	I	0x13	0x6		rd = rs1 imm	
andi	And Immediate	I	0x13	0x7		rd = rs1 & imm	
slli	Shift Left Logical Imm	I	0x13	0x1	imm[11:6]=0x00	rd = rs1 << imm[5:0]	
srli	Shift Right Logical Imm	I	0x13	0x5	imm[11:6]=0x00	rd = rs1 >> imm[5:0]	
sraiw	Shift Right Arith Imm	I	0x13	0x5	imm[11:6]=0x10	rd = rs1 >> imm[5:0]	
slti	Set Less Than Imm	I	0x13	0x2		rd = (rs1 < imm)?1:0	
sltiu	Set Less Than Imm (U)	I	0x13	0x3		rd = (rs1 < imm)?1:0	
lb	Load Byte	I	0x03	0x0		rd = M[rs1+imm][7:0]	
lh	Load Half	I	0x03	0x1		rd = M[rs1+imm][15:0]	
lw	Load Word	I	0x03	0x2		rd = M[rs1+imm][31:0]	
ld	Load Double Word	I	0x03	0x3		rd = M[rs1+imm][63:0]	RV64I
lbu	Load Byte (U)	I	0x03	0x4		rd = M[rs1+imm][7:0]	
lhu	Load Half (U)	I	0x03	0x5		rd = M[rs1+imm][15:0]	
lwu	Load Word (U)	I	0x03	0x6		rd = M[rs1+imm][63:0]	RV64I
sb	Store Byte	S	0x23	0x0		M[rs1+imm][7:0] = rs2[7:0]	
sh	Store Half	S	0x23	0x1		M[rs1+imm][15:0] = rs2[15:0]	
sw	Store Word	S	0x23	0x2		M[rs1+imm][31:0] = rs2[31:0]	
sd	Store Double Word	S	0x23	0x3		M[rs1+imm][63:0] = rs2[63:0]	RV64I
beq	Branch ==	B	0x63	0x0		if(rs1 == rs2) PC += imm	
bne	Branch !=	B	0x63	0x1		if(rs1 != rs2) PC += imm	
blt	Branch <	B	0x63	0x4		if(rs1 < rs2) PC += imm	
bge	Branch ≥	B	0x63	0x5		if(rs1 ≥ rs2) PC += imm	
bltu	Branch < (U)	B	0x63	0x6		if(rs1 < rs2) PC += imm	
bgeu	Branch ≥ (U)	B	0x63	0x7		if(rs1 ≥ rs2) PC += imm	
jal	Jump And Link	J	0x6F			rd = PC+4; PC += imm	
jalr	Jump And Link Register	I	0x67	0x0		rd = PC+4; PC = rs1 + imm	
lui	Load Upper Imm	U	0x37			rd = imm << 12	
auipc	Add Upper Imm to PC	U	0x17			rd = PC + (imm << 12)	
ecall	Environment Call	I	0x73	0x0	imm=0x0	Appel système	
ebreak	Environment Break	I	0x73	0x0	imm=0x1	Appel au débogueur	
mul	Multiply	R	0x33	0x0	0x01	rd = (rs1 * rs2)[31:0]	RV32M
mulh	Multiply High	R	0x33	0x1	0x01	rd = (rs1 * rs2)[63:32]	RV32M
mulhsu	Multiply High (S) (U)	R	0x33	0x2	0x01	rd = (rs1 * rs2)[63:32]	RV32M
mulhu	Multiply High (U)	R	0x33	0x3	0x01	rd = (rs1 * rs2)[63:32]	RV32M
div	Divide	R	0x33	0x4	0x01	rd = rs1 / rs2	RV32M
divu	Divide (U)	R	0x33	0x5	0x01	rd = rs1 / rs2	RV32M
rem	Remainder	R	0x33	0x6	0x01	rd = rs1 % rs2	RV32M
remu	Remainder (U)	R	0x33	0x7	0x01	rd = rs1 % rs2	RV32M

Format des instructions

	31	25	24	20	19	15	14	12	11	7	6	0
R	fn7			rs2	rs1	fn3	rd			opcode		
I	imm[11:0]				rs1	fn3	rd			opcode		
S	imm[11:5]			rs2	rs1	fn3	imm[4:0]			opcode		
B	imm[12 10:5]			rs2	rs1	fn3	imm[4:1 11]			opcode		
U	imm[31:12]						rd			opcode		
J	imm[20 10:1 11 19:12]						rd			opcode		

Note: petit-boutiste et aligné sur 4 octets

Registres

Registre	Nom d'ABI	Description	Qui sauve ?
x0	zero	Constante zero	—
x1	ra	Return address	Appellant
x2	sp	Stack pointer	Appelé
x3	gp	Global pointer	—
x4	tp	Thread pointer	—
x5-x7, x28-x31	t0-t6	Temporaires	Appellant
x8, x9, x18-x27	s0-s11	Sauvegardés	Appelé
x10-x17	a0-a7	Arguments (et retours)	Appellant

