

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 1**

NOM : ..... PRENOM : ..... SECTION : .....

**Consignes à lire impérativement !**

L'examen est composé de **2 parties**, chacune d'une durée de **2 heures**. Veuillez respecter les consignes suivantes.

1. Calculatrice, téléphone portable et notes de cours ne sont pas autorisés. Laissez-les dans votre sac. Pensez à éteindre votre téléphone portable !
2. Ecrivez vos **nom, prénom** et **section** (math, info, ...) sur chaque feuille, y compris les feuilles de brouillon.
3. Faites attention à la clarté et à l'organisation de vos réponses. Respectez les règles grammaticales et orthographiques.
4. Utilisez pour vos réponses les **cadres** prévus à cet effet. Si davantage d'espace est nécessaire, utilisez le dos de la feuille ou une feuille supplémentaire et indiquez clairement où se situe le restant de la réponse.
5. Vous devez terminer cette partie de l'examen avant de pouvoir sortir de la salle (pour aller aux toilettes par exemple).
6. Vérifiez que vous avez répondu à toutes les questions.
7. Toutes les feuilles doivent être remises en fin d'examen, y compris les brouillons.

**Question 1 – Echauffement (/5)**

**Q1a – Un système informatique est doté d'une mémoire principale (DRAM) ayant un temps d'accès de 50 ns. On considère l'ajout d'une cache (SRAM) ayant un temps d'accès de 10 ns. Quel doit être le *hit ratio* minimum pour que l'ajout de cette cache réduise le temps d'accès moyen à la mémoire ?**

- ☐ 1 %.
- ☐ 5 %.
- ☐ 10 %.
- ☐ 20 %.
- ☐ 33 %.
- ☐ Aucune des réponses ci-dessus.

**Justification :** .....  
.....  
.....

# Examen du cours de Fonctionnement des Ordinateurs

## 1<sup>ère</sup> Session, Juin 2023

### Partie 1

NOM : ..... PRENOM : ..... SECTION : .....

**Q1e – On suppose un système de mémoire virtuelle ayant les paramètres suivants. La mémoire physique contient 512 MB. L'espace d'adressage virtuel a une taille de 4 GB. Les mémoires physique et virtuelle sont organisées en pages qui ont une taille fixe de 8 KB. Veuillez calculer les éléments suivants :**

- Taille en bits d'un offset dans une page : .....
- Taille en bits d'un numéro de page virtuel (VPN) : .....
- Taille en bits d'un numéro de page physique (PPN) : .....
- Nombre d'entrées de la table des pages : .....
- Taille en bits d'une entrée de la table des pages : .....
- Taille en octets de la table des pages : .....

**Q1d – Dans l'architecture MIPS, le mot binaire 0x10A0FFEF est le code d'une instruction BEQ qui teste l'égalité des registres 5 (\$a1) et 0 (\$zero) et effectue un branchement relatif le cas échéant. Ce code d'instruction comporte une valeur immédiate dans ses 16 bits de poids faible (type I, illustré à la Figure 1). Dans l'instruction BEQ, cette valeur immédiate sert à calculer le décalage du *Program Counter* (PC). Si PC vaut 0x0400006C lors de l'exécution de l'instruction BEQ, que vaudra PC après l'exécution de l'instruction**

- si les registres testés sont différents ? .....
- si les registres testés sont égaux ? .....

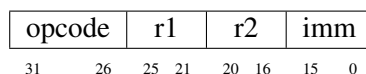


FIGURE 1 – Format d'une instruction de type I. Les nombres affichés sous les différentes parties de l'instruction indiquent leurs premier et dernier bits.

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 1**

NOM : ..... PRENOM : ..... SECTION : .....

**Q1b – Exécutez mentalement ou sur papier le programme MIPS donné ci-dessous. Fournissez la sortie affichée à la console par le programme.**

.....  
.....  
.....  
.....  
.....

```
1      .text
2 main:
3      la    $a0, data
4      lw    $a1, 0($a0)
5 there:
6      andi  $a0, $a1, 1
7      srl   $a1, $a1, 1
8      addi  $a0, $a0, '0'
9      li    $v0, 11
10     syscall
11     bne   $a1, $zero, there
12     jr    $ra
13
14 data:
15     .word 101
16
17 # Notes:
18 # - l'appel systeme 11 affiche a la console le caractere dont le code est dans $a0.
19 # - la directive .word place l'entier qui suit en memoire sur 32 bits.
```

**Q1c – Illustrez à l'aide d'un programme d'assemblage en langage MIPS contenant 3 instructions l'intérêt de l'adressage indirect indexé.**

.....  
.....  
.....

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 1**

NOM : ..... PRENOM : ..... SECTION : .....

**Question 2 – Représentation de nombres à virgule flottante (/5)**

On considère une représentation de nombre flottants qui suit les principes d'IEEE 754, mais en adoptant des tailles de mantisse et d'exposant différentes. On note le bit de signe  $s$ , l'exposant  $e$  et la mantisse  $m$ . On désigne par  $E$  la taille (en bits) de l'exposant et par  $M$  la taille (en bits) de la mantisse. Les valeurs de  $E$  et  $M$  considérées dans cette question sont données à la Table 1. Le calcul du biais  $B$  ainsi que la distinction entre représentations normalisée, dénormalisée et valeurs spéciales reposent sur les mêmes conventions qu'avec IEEE 754. Les arrondis sont calculés selon *round-to-nearest-even*.

$E$	3
$M$	6

TABLE 1 – Valeurs des paramètres  $E$  et  $M$ .

**Ce qui vous est demandé :** Veuillez répondre à toutes les questions ci-dessous, en justifiant vos réponses. L'absence de justification est équivalente à l'absence de réponse.

**Q2a – Donnez la valeur du biais  $B$  du système.**

.....  
.....

**Q2b – Donnez la valeur de l'épsilon machine  $\epsilon_M$ .**

.....  
.....

**Q2c – Déterminez le nombre représenté par  $s = 0$ ,  $e = 001$  et  $m = 110001$ .**

.....  
.....

**Q2d – Déterminez le nombre représenté par  $s = 1$ ,  $e = 000$  et  $m = 101100$ .**

.....  
.....

**Q2e – Déterminez le nombre représenté par  $s = 0$ ,  $e = 111$  et  $m = 000000$ .**

.....  
.....

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 1**

NOM : ..... PRENOM : ..... SECTION : .....

**Q2f – Donnez la représentation binaire de  $x = 0,09$ .**

.....

.....

.....

.....

**Q2g – Donnez la représentation binaire de  $y = 1,191$ .**

.....

.....

.....

.....

**Q2h – En utilisant leur représentation binaire, effectuez l'addition des nombres  $x$  et  $y$  donnés ci-dessus.**

.....

.....

.....

.....

.....

.....

**Q2i – En utilisant leur représentation binaire, effectuez la soustraction des nombres  $x$  et  $y$  donnés ci-dessus.**

.....

.....

.....

.....

.....

.....

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 2**

NOM : ..... PRENOM : ..... SECTION : .....

**Consignes à lire impérativement !**

L'examen est composé de **2 parties**, chacune d'une durée de **2 heures**. Veuillez respecter les consignes suivantes.

1. Calculatrice, téléphone portable et notes de cours ne sont pas autorisés. Laissez-les dans votre sac. Pensez à éteindre votre téléphone portable !
2. Ecrivez vos **nom**, **prénom** et **section** (math, info, ...) sur chaque feuille, y compris les feuilles de brouillon.
3. Faites attention à la clarté et à l'organisation de vos réponses. Respectez les règles grammaticales et orthographiques.
4. Utilisez pour vos réponses les **cadres** prévus à cet effet. Si davantage d'espace est nécessaire, utilisez le dos de la feuille ou une feuille supplémentaire et indiquez clairement où se situe le restant de la réponse.
5. Vous devez terminer cette partie de l'examen avant de pouvoir sortir de la salle (pour aller aux toilettes par exemple).
6. Vérifiez que vous avez répondu à toutes les questions.
7. Toutes les feuilles doivent être remises en fin d'examen, y compris les brouillons.

**Question 1 – Mapping de tableau en MIPS (/7)**

Etant données deux séquences de même longueur,  $\mathbf{a} = (a_0, \dots, a_{n-1})$  et  $\mathbf{b} = (b_0, \dots, b_{n-1})$ , on souhaite produire une troisième séquence  $\mathbf{c} = (c_0, \dots, c_{n-1})$  en utilisant une fonction  $f$  appliquée à des paires d'éléments issus de  $\mathbf{a}$  et  $\mathbf{b}$ . Les éléments de la séquence  $\mathbf{c}$  sont générés comme montré à l'Equation (1). Par exemple, si  $\mathbf{a} = (1, 2, 3)$  et  $\mathbf{b} = (2, 1, 5)$  et  $f(x, y) = x + y$ , alors  $\mathbf{c} = (3, 3, 8)$ . Si  $f(x, y) = 2x + y^2$ , alors la séquence résultante est  $\mathbf{c} = (6, 5, 31)$ .

$$\forall 0 \leq k < n, c_k = f(a_k, b_k) \quad (1)$$

L'objectif de la question est d'effectuer la génération de la séquence  $\mathbf{c}$  à l'aide d'une fonction  $\text{map}(\mathbf{a}, \mathbf{b}, \mathbf{c}, n, f)$  implémentée en langage d'assemblage MIPS. Celle-ci prend 5 arguments et ne retourne pas de résultat.

- Les arguments  $\mathbf{a}$ ,  $\mathbf{b}$  et  $\mathbf{c}$  désignent les séquences à manipuler,  $\mathbf{a}$  et  $\mathbf{b}$  étant les séquences source et  $\mathbf{c}$  la séquence à générer. Chaque séquence est implémentée sous forme d'un tableau dont les éléments sont des naturels représentés sur 32 bits.  $\mathbf{a}$ ,  $\mathbf{b}$  et  $\mathbf{c}$  sont les adresses de ces tableaux.
- L'argument  $n$  est la taille des tableaux. Cette taille est commune aux trois tableaux.
- L'argument  $f$  est l'adresse de la fonction à appliquer pour produire les éléments de la séquence résultat. Celle-ci doit respecter la convention suivante : elle prend deux arguments  $x$  et  $y$  de type naturel sur 32 bits et retourne un résultat de même type.

La fonction  $\text{map}$  se charge d'appeler  $n$  fois la fonction  $f$  pour générer chacun des éléments  $c_k$  à partir d'une paire d'éléments  $a_k$  et  $b_k$ , comme montré à l'Equation (1).

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 2**

NOM : ..... PRENOM : ..... SECTION : .....

**Ce qui vous est demandé :** vous devez fournir les éléments suivants :

1. L'implémentation de la fonction `map`, conformément à la spécification ci-dessus.
2. L'implémentation de la fonction `fct_poly(x, y) = 2x + y2`.
3. La déclaration de 3 tableaux de naturels représentés sur 32 bits : `tabA`, `tabB` et `tabC` de même longueur  $n = 5$ . Les éléments des tableaux `tabA` et `tabB` sont respectivement  $(0, 1, 2, 3, 4)$ ,  $(1, 3, 5, 7, 9)$ . Le tableau `tabC` doit se voir allouer de l'espace en mémoire mais son contenu ne doit pas nécessairement être initialisé.
4. Un programme principal (`main`) qui appelle la fonction `map` afin d'appliquer la fonction `fct_poly` aux tableaux `tabA` et `tabB` afin de générer le tableau `tabC`. Il est à noter que la fonction `map` se charge elle-même d'appeler la fonction `fct_poly` pour générer chaque élément de `tabC`.

La convention d'appel suivante doit être respectée par vos fonctions. Les 4 premiers arguments sont passés dans les registres `a0` à `a3`. Les arguments excédentaires sont passés sur la pile. Si une fonction retourne un résultat, elle utilise le registre `v0` à cet effet.

**Q1a – Fonction `fct_poly`**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 2**

NOM : ..... PRENOM : ..... SECTION : .....

### Q1b – Fonction map



**Examen du cours de Fonctionnement des Ordinateurs**  
1<sup>ère</sup> Session, Juin 2023  
**Partie 2**

NOM : ..... PRENOM : ..... SECTION : .....

### Q1c – Déclaration des tableaux `tabA`, `tabB` et `tabC` et fonction `main`

**Examen du cours de Fonctionnement des Ordinateurs**  
1<sup>ère</sup> Session, Juin 2023  
**Partie 2**

NOM : ..... PRENOM : ..... SECTION : .....

**Question 2 – Performance de la mémoire cache (/3)**

Cette question s'intéresse à l'impact du type et des caractéristiques d'une cache sur les performances atteintes par un programme. A cette fin, on considère le programme MIPS de la Figure 2. En exécutant pas à pas les instructions de ce programme, on souhaite simuler le comportement de la cache et déterminer quels accès à la cache sont des succès (*hit*) ou des échecs (*miss*) pour *in fine* quantifier le *miss rate*. On considère que des caches séparées sont utilisées pour les données (D-Cache) et pour les instructions (I-Cache). On ne s'intéresse qu'aux accès à la D-Cache.

```

1      addi $t0, $zero, 5
2  loop:
3      beq $t0, $zero, done
4      lw $t1, 0x4($zero)
5      lw $t2, 0x24($zero)
6      addi $t0, $t0, -1
7      j loop
8  done:

```

FIGURE 2 – Programme MIPS.

Deux architectures différentes de caches sont considérées : CACHE 1 est de type *direct-mapped* tandis que CACHE 2 est *2-way-associative*. Elles sont toutes les deux d'une capacité de 32 octets et organisées en 8 lignes de 4 octets.

**Il vous est demandé** de fournir les informations suivantes pour l'exécution du programme sur les deux caches.

1. Indiquez le comportement des 2 caches pour les 5 premiers accès mémoire effectués par le programme.
2. Structure d'une adresse : nombre de bits d'offset, de set et de tag.
3. Nombre d'accès se traduisant par un *miss* – distinguer entre *compulsory*, *capacity* et *conflict misses* ;
4. Proportion de misses (*miss ratio*).

**Q2a – Trace des 5 premiers accès (adresse, tag/set/offset, *hit* ou *miss*)**

Adresse	CACHE 1	CACHE 2

**Examen du cours de Fonctionnement des Ordinateurs**  
**1<sup>ère</sup> Session, Juin 2023**  
**Partie 2**

NOM : ..... PRENOM : ..... SECTION : .....

**Q2b (CACHE 1 : *direct-mapped*)**

Structure d'adresse	
<i>bits d'offset</i>	
<i>bits de set</i>	
<i>bits de tag</i>	
Performance	
Nombre de <i>misses</i>	
<i>compulsory</i>	
<i>conflict</i>	
<i>capacity</i>	
<i>Miss ratio</i>	

**Q2c (CACHE 2 : *2-way-associative*)**

Structure d'adresse	
<i>bits d'offset</i>	
<i>bits de set</i>	
<i>bits de tag</i>	
Performance	
Nombre de <i>misses</i>	
<i>compulsory</i>	
<i>conflict</i>	
<i>capacity</i>	
<i>Miss ratio</i>	