Question **1**

Réponse enregistrée

Noté sur 19

Remarques concernant le codage des réponses

Les digits hexadécimaux supérieurs à 9 sont représentés par les caractères A jusqu'à F en majuscule.

Les nombres hexadécimaux contiennent **obligatoirement** 8 caractères hexadécimaux. (pas d'espace ni d'autres caractères)

Les nombres décimaux contiennent uniquement des chiffres de 0 à 9.

Les nombres binaires contiennent uniquement des 0 et des 1.

1) Nombres entiers : Donnez les valeurs hexadécimales 32 bits des expressions suivantes:

0xFEDCBA98 & 0x89ABCDEF = 0x 88888888

0xFEDCBA98 | 0x89ABCDEF = 0x FFFFFFFF

0xFEDCBA98 ^ 0x89ABCDEF = 0x 77777777

0xFEDCBA98 SRL **4** = **0x** 0FEDCBA9

-119 SRA **4** = **0x** FFFFFFF8

2) Nombre réel

- Donnez la valeur **binaire** normalisée représentant le nombre réel -119,25 =**1,** 11011101 2^ 6 . **Attention: La réponse se termine par 1.**

- Donnez la valeur **décimale** de l'exposant codé au format IEEE 754 simple précision pour le nombre réel -119,25 = 133

- Donnez la valeur **hexadécimale** représentant le nombre réel -119,25 au format IEEE 754 simple précision **0x** C2EE8000

3) Soit la fonction foo1

```
int foo1(int v,int b){ // b 0 ou 1 uniquement
    return (v ^ (-b)) + b;
}
```

- Donnez la valeur **décimale** résultat de l'évaluation de foo1(-119,0)= -119

- Donnez la valeur **décimale** résultat de l'évaluation de foo1(-119,1)= 119

4) Complétez le code C de la fonction permettant de compter le nombre de bit à 0 dans une donnée 32 bits.

Les réponses ne contiendront pas d'espace ni de parenthèse.

```
unsigned countNot(unsigned n){
    unsigned res= 32 ;
    while ( n!=0 ) { // contient une variable, un opérateur relationnel et une constante (sans espace).
        res=res-( n&1 ); // une variable un opérateur bit à bit et une constante (sans espace)
        n=n >> 1;
    }
    return res;
}
```

5) Soit la fonction foo2

```
void foo2(unsigned  *x){
    unsigned var1;
    var1 = countNot(*x);
    do{
        *x=*x+1;
    }while(countNot(*x)!=var1);
}
```

- Quelle est la valeur **décimale** n après l'évaluation de foo2 dans le cas suivant:

unsigned n=119; foo2(&n); n= 123

-Quelles sont les valeurs **hexadécimales** de n après l'évaluation de foo2 dans les cas suivants:

unsigned n=0xFFFFFFFD; foo2(&n); n= **0x** FFFFFFFE

unsigned n=0xFFFFFFFE; foo2(&n); n= **0x** 7FFFFFFF

Question 1

Réponse enregistrée

Noté sur 20

Complétez le code assembleur des fonctions précédentes (countNot et foo2)

```
.data
    nbr : .word 119
.text
    la $a0,--1--
    jal foo2
    lw --2--,0($a0)
    li $v0,--3--
    syscall # Afficher resultat foo2
    li $v0,--4--
    syscall # Terminer
```

--1-- nbr

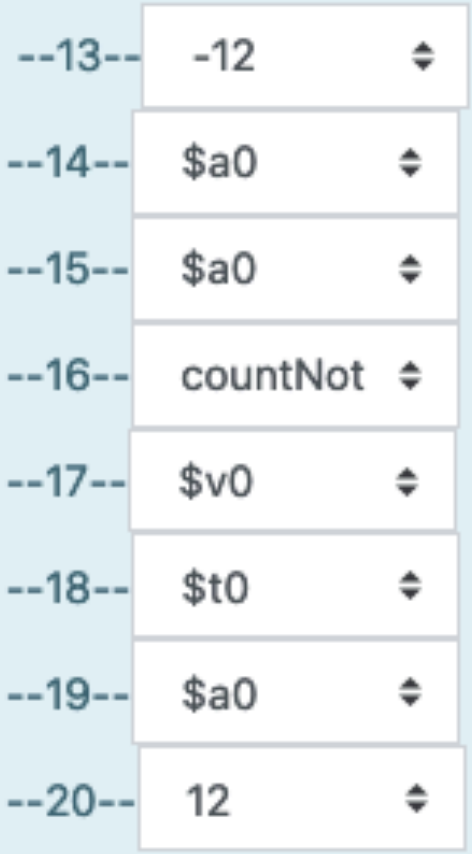
--2-- $a0

--3-- 1

--4-- 10

```
countNot:
# $a0 <- n # $v0 <- nombre bit a 0
     sw $a0,--5--($sp)
     sw $t0,--6--($sp)
     --7-- $v0,32
countNotBoucle:
     beq --8--,$0,countNotFin
     andi $t0,--9--,--10--
     --11-- $v0,$v0,$t0
     --12-- $a0,$a0,1
     b countNotBoucle
countNotFin:
     lw $a0,0($sp)
     lw $t0,-4($sp)
     jr $ra
```

```
foo2:
# a0 : donnée - résultat
     sw $a0,0($sp)
     sw $ra,-4($sp)
     sw $t0,-8($sp)
     addi $sp,$sp,--13--
     lw --14--,0(--15--)
     jal --16--
     move $t0,--17--
foo2Boucle:
     addi $a0,$a0,1
     jal countNot
     bne $v0,--18--,foo2Boucle
     move $t0,--19--
     lw $a0,--20--($sp)
     sw $t0,0($a0)
foo2Fin:
     addi $sp,$sp,12
     lw $a0,0($sp)
     lw $ra,-4($sp)
     lw $t0,-8($sp)
     jr $ra
```



Question **1**

Réponse enregistrée

Noté sur 10

1)Soit le programme suivant

```
1   .data
2       source : .asciiz "Hello_world!!!!"
3                : .space 256
4   .text
5       la $a0,source
6   test:
7       lb $t0,0($a0)
8       sb $t0,64($a0)
9       addi $a0,$a0,1
10      bne $t0,$0,test
11      li $v0,10
12      syscall
```

Soit un cache pédagogique contenant 64 octets, organisé par bloc de 8 octets. Nous souhaitons évaluer les performances du programme ci-dessus. Nous évaluerons les fonctions de correspondance directe et associative par ensemble de deux blocs.

1) Combien d'itérations sont effectuées 16

2) Quelle est la taille en bit du **déplacement** 3

3) Quelle est la taille en bit de l'**index** dans le cas directe 3 et dans le cas associative 2

3) Quelle est la taille en bit de l'**étiquette** dans le cas directe 26 et dans le cas associative 27

4) Donnez le **taux de succès** dans le cas directe 0 /32 et dans le cas associative 28 /32

Question 1

Réponse enregistrée

Noté sur 8

Soit le programme suivant

```
test:
        lb $t0,0($a0)
        sb $t0,64($a0)
        addi $a0,$a0,1
        bne $t0,$0,test
        li $v0,10
        syscall
```

L'instruction lb $t0,0($a0) commence au cycle 1 et se termine au cycle 5.

Sans optimisation

- A quel cycle commence l'instruction sb $t0,64($a0) 2
- A quel cycle commence l'instruction addi $a0,$a0,1 3
- A quel cycle commence l'instruction bne $t0,$0,test 6
- A quel cycle commence le deuxième traitement de l'instruction lb $t0,0($a0) 10

Avec envoi et branchement au plus tôt

- A quel cycle commence l'instruction sb $t0,64($a0) 2
- A quel cycle commence l'instruction addi $a0,$a0,1 3
- A quel cycle commence l'instruction bne $t0,$0,test 4
- A quel cycle commence le deuxième traitement de l'instruction lb $t0,0($a0) 6

Question **1**

Réponse enregistrée

Noté sur 5

1)Soit le programme suivant:

```
test:
    lb $t0,0($a0)
    sb $t0,64($a0) # instruction 1
    addi $a0,$a0,1
    bne $t0,$0,test #instruction 2
```

-Donnez la valeur hexadécimale représentant l'**instruction 1** 0x A0880040

-Donnez la valeur hexadécimale représentant l'**instruction 2** 0x 1500FFFC

Attention : les digits hexadécimaux supérieurs à 9 sont en majuscule

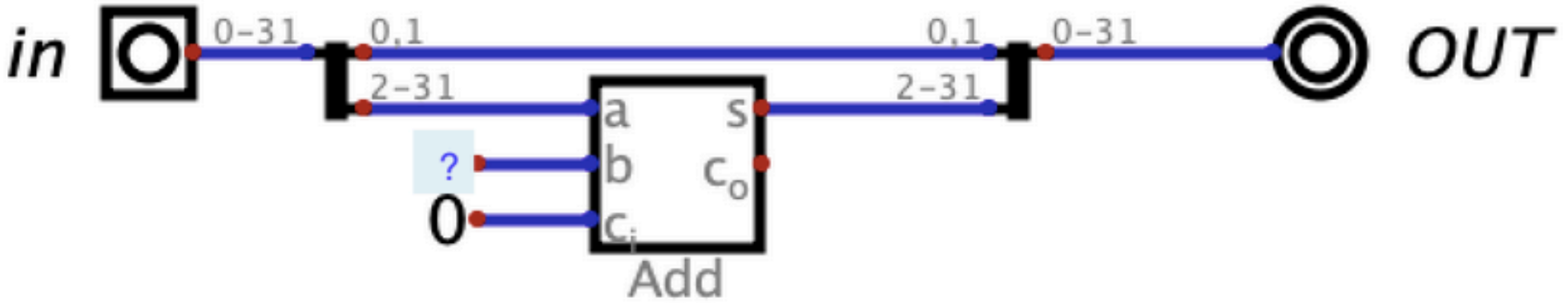
2)Soit le programme suivant:

```
lui $t0,0x1001
lw  $t1,0($t0)
sw  $t1,1($t0)
```

Quelle instruction provoque une exception sw

Donnez la valeur des bits 2 à 6 du registre $13 5

3)Nous souhaitons créer un circuit plus rapide permettant d'ajouter 4 à la donnée entrante.



Quelle valeur devons nous envoyer sur l'entrée **b** de l'additionneur 1