

TD ASR2 (Programmation assembleur) 1ère Année

Questionnaire à choix multiples – Révision et application des notions de base

1) D'après-vous, pourquoi l'optimiseur du compilateur C essaie-t-il de remplacer les instructions de branchement (saut) par des instructions équivalentes ?

- A-pour gagner de la place en mémoire
- B-parce qu'un branchement est une instruction qui s'exécute "lentement"
- C-pour éviter de "casser" la file d'attente du "pipeline"

2) A quoi sert le registre de liens (Link Register) :

- A-à conserver l'adresse de retour dans le programme d'appel, lors d'un appel de sous-programme
- B-à éviter un branchement incondionnel
- C-à faciliter l'édition des liens (liaison dynamique)

3) Dans l'instruction "stw r31,8(r1)", l'adressage du deuxième opérande est de type:

- A-indirect
- B-direct
- C-immédiat

4) En exécutant la séquence suivante:

```
li r3,2
xor r4,r4,r4
cmpw cr1,r4,r3
bgt cr1,ETIQUETTE
...
```

- A-le programme ne signale aucune erreur et continue l'exécution en séquence (après l'instruction de branchement)
- B-le programme ne signale aucune erreur et continue l'exécution à l'adresse ETIQUETTE
- C-le programme signale une erreur lors de l'exécution de l'instruction de branchement

5) Le fragment de programme suivant :

```
cmpwi cr2,r7,0
bge cr2,suite
nand r7,r7,r7
addi r7,r7,1
```

suite:

...

A- incrémente le registre 7 de 1 seulement si celui-ci est positif ou nul

B- remplace le registre 7 par son complément à deux incrémenté de 1 seulement si le registre 7 est positif ou nul

C- remplace le contenu du registre 7 par sa valeur absolue.

6) Le fragment de programme suivant :

etiq :

```
...
andi r0,r9,1
beq cr0,etiq
```

...

A- se poursuit à l'adresse logique "etiq" seulement si le registre 9 contient 0

B- se poursuit en séquence si le registre 9 contient une valeur impaire

C- se poursuit en séquence si le registre 9 contient une valeur paire

7) La séquence suivante :

```
cmpw cr0,r1,r2
bgt cr0,etiq
mr r3,r2
b suite
etiq: mr r3,r1
suite:
```

...

A- permet de remplacer le contenu du registre 3 par le "maximum" entre R1 et R2

B- permet de remplacer le contenu du registre 3 par le "minimum" entre R1 et R2

C- permet d'échanger les contenus de R1 et R2

8) Le registre 9 contient 10 en base 10. Que contient le registre 3 après exécution de la séquence :

```
srawi r3,r9,1
addze r3,r3
```

A- 6 en base 10

B- 7 en base 10

C- 5 en base 10

9) Pour récupérer dans RT le reste de la division entière de RA par RB, on peut utiliser la séquence suivante :

A- `divw RT,RB,RA`
`mullw RT,RT,RA`
`subf RT,RT,RB`

B- `divw RT,RA,RB`
`mullw RT,RT,RB`
`subf RT,RT,RA`

C- `divw RT,RA,RB`
`mullw RT,RT,RB`
`subf RT,RA,RT`

10) La séquence suivante d'instructions :

```
xor r1,r1,r2
xor r2,r1,r2
xor r1,r1,r2
```

A- permet d'échanger les contenus des registres 1 et 2

B- permet de mettre à zéro les registres 1 et 2

C- permet de complémenter les contenus des registres 1 et 2.