

**IFT3380/6380 : Architecture des ordinateurs contemporains  
Examen intra (Hiver 2004)**

**Professeur:** El Mostapha Aboulhamid

Directives

- Toute documentation est permise.
- Une calculatrice (même programmable) est permise.
- Le barême est indiqué entre parenthèses.
- Inscrire votre nom et code permanent sur le questionnaire.
- Si vous ne comprenez pas exactement une question, faites une interprétation, expliquez la et continuez. Veillez à la clarté de vos réponses et justifiez les.
- *Les questions sont indépendantes et leur ordre n'indique pas leur degré de difficulté.*


**NOM:****Code permanent:****Question 1 (25)**

On suppose une machine DLX où le branchement est retardé avec un créneau d'attente d'une instruction et que les branchements conditionnels sont résolus à l'étape 2 (étape décodage). On suppose aussi que la latence entre une instruction productrice et une consommatrice est comme suit :

Instruction productrice	Instruction consommatrice	Latence
Op UAL Virgule flottante	Op UAL V.F.	1
Op UAL V.F.	Stockage double en mémoire	3
Chargement double de la mémoire	Op UAL V.F.	2
Chargement double	Stockage double	1

Soit le fragment de code suivant:

```
BOUCLE: LD      F0, 0(R1)
         MULTD  F0, F0, F2
         LD      F4, 0(R2)
         ADDD   F0, F0, F4
         SD      0(R2), F0
         SUBI   R2, R2, 8
         SUBI   R1, R1, 8
         BNEZ  R1, BOUCLE
         NOP
```

1. Récrire le code en explicitant les suspensions qui existent.

**NOM:** \_\_\_\_\_

**Code permanent:** \_\_\_\_\_

2. Dérouler la boucle une fois et l'ordonner afin de minimiser les suspensions ('stalls'). Les tests intermédiaires sur R1 doivent être ignorés.

3. Combien de fois faut-il dérouler la boucle afin de permettre un ordonnancement qui élimine toutes les suspensions (nombre de copies d'itérations)? On suppose que les tests intermédiaires sur R1 sont ignorés.



NOM: \_\_\_\_\_

Code permanent: \_\_\_\_\_

**Question 3 (25)**

Soit une machine pipeline, ayant 5 étages (IF, ID, EX, MEM, WB). Les branchements inconditionnels sont résolus à la fin de l'étage ID alors que les branchements conditionnels ne le sont qu'à la fin de l'étage EX. On suppose la distribution suivantes: 30% des instructions sont des branchements conditionnels (dont 60% sont pris) et 10% sont des sauts inconditionnels ou des appels de sous-programmes.

1. Combien de suspensions ('stalls') on a
  - a. pour un branchement conditionnel? \_\_\_\_\_
  - b. pour un branchement inconditionnel? \_\_\_\_\_
2. Calculer le CPI (cycles par instruction) de cette machine, si on ne fait ni prédiction ni remplissage du créneau d'attente.
3. On suppose que les branchements, qu'ils soient conditionnels ou non, sont différés avec un créneau d'attente. On suppose que ce créneau est rempli correctement pour 30% des branchements. Calculer le nouveau CPI.

**Question 4 (25)**

Soit l'exemple ci-après similaire à celui vu dans le cours pour illustrer l'ordonnancement dynamique d'une boucle de traitement suivant l'algorithme de Tomasulo. On suppose que les chargements de mémoire prennent 1 unité de temps, les stockages 3 unités, l'addition virgule flottante 1 unité et la multiplication 4.

1. Compléter la figure ci-dessous pour indiquer l'état de la machine au cycle 10.

ITER	Instruction	<i>j</i>	<i>k</i>	Issue	Comp	Result
1	LD	F0	0	R1		
1	ADDD	F2	F0	F2		
1	MULTD	F4	F0	F2		
1	SD	F4	0	R1		
2	LD	F0	0	R1		
2	ADDD	F2	F0	F2		
2	MULTD	F4	F0	F2		
2	SD	F4	0	R1		

	Busy	Addr	Fu
Load1			
Load2			
Load3			
Store1			
Store2			
Store3			

*Reservation Stations:*

Time	Name	Busy	Op	Vj	Vk	S1	S2	RS
						Qj	Qk	
	Add1							
	Add2							
	Add3							
	Mult1							
	Mult2							

*Code:*

LD	F0	0	R1
ADDD	F2	F0	F2
MULTD	F4	F0	F2
SD	F4	0	R1
SUBI	R1	R1	#8
BNEZ	R1	Loop	

*Register result status*

Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
0	80									

N.B. : Pour le stockage en mémoire, une opération de stockage même si elle est lancée si une place est disponible dans la station de réservation ne commencera son étape d'exécution que si l'opération de stockage précédente a terminé son étape d'exécution.