

**Examen semestriel**  
**Architecture des ordinateurs (Documents non autorisés)**

**Exo 02 :** Traduire en assembleur MIPS le programme C suivant:

```
if (a < b) a = 2*(a-b) + c*c ;  
else      c = a /b
```

**Exo 03 :** Écrire un Programme en assembleur MIPS permettant de déterminer le nombre d'occurrence d'une valeur **X** dans un tableau '**tab**' d'entiers de taille '**T**':

```
Exemple : Tab = { 4, -5, 2, 4, 3}, T = 5, X = 4,  
Résultat = 2
```

**Exo 04 :** Coder les instructions MIPS suivantes (Résultats en nHéxa):

<b>sub</b> \$t1, \$t1, \$t0	
<b>bne</b> \$v0, \$v1, plus            # plus = 0x A00B	
<b>lb</b> \$t0, -10(\$sp)	
<b>0x34300000</b>	
<b>0x8E0BFFF6</b>	
<b>0x0000000D</b>	

**Exo 05 :** On veut résoudre une équation de second degré  $a*x^2 + b*x + c$  tels que a, b et c sont des entiers

Questions :

- 1- Ecrire une fonction **Delta(a,b,c)** permettant de calculer le discriminant  $\Delta = b^2 - 4a*c$  et retourner le résultat dans \$V0.
- 2- Ecrire un programme main permettant de résoudre une équation de second degré :
  - a, b et c doivent être lu au clavier (des entiers)
  - On suppose qu'on a une fonction **Sqrt(X)** permettant de calculer la racine carrée d'un entier X et retourne le résultat dans \$V0
  - Tous les nombres manipulés sont supposés être des entiers

Tournez la page SVP ../..

Exo supplémentaire : Soit le programme MIPS suivant :

	lw	\$t0, 16(\$sp)
	lw	\$t1, 12(\$sp)
toto:	beq	\$t0, \$t1, Etoto
	bge	\$t0, \$t1, etiq1
	sub	\$t1, \$t1, \$t0
	j	etiq2
etiq1:	sub	\$t0, \$t0, \$t1
etiq2:		
	j	toto
Etoto:	move	\$v0, \$t0

**Question :** Quelle est la valeur finale de \$v0 à la fin de l'exécution en supposant que \$t0 = 10 et \$t1 = 20 à l'état initial.

---

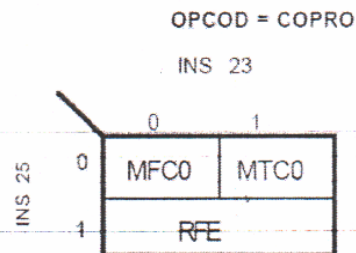
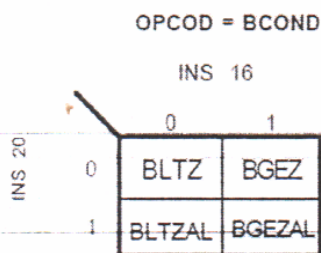
Bon courage

31	25	20	15	10	5	0	Instr
0	RS	RT	RD	SA	0		sll
0	RS	RT	RD	SA	2		srl
0	RS	RT	RD	SA	3		sra
0	RS	RT	RD	0	4		sllv
0	RS	RT	RD	0	6		srlv
0	RS	RT	RD	0	7		srav
0	RS	0			8		jr
0	RS	0	RD	0	9		jalr
0						12	syscall
0						13	break
0	0		RD	0	16		mfhi
0	RS	0			17		mthi
0	0		RD	0	18		mflo
0	RS	0			19		mtlo
0	RS	RT	0		24		mult
0	RS	RT	0		25		multu
0	RS	RT	0		26		div
0	RS	RT	0		27		divu
0	RS	RT	RD	0	32		add
0	RS	RT	RD	0	33		addu
0	RS	RT	RD	0	34		sub
0	RS	RT	RD	0	35		subu
0	RS	RT	RD	0	36		and
0	RS	RT	RD	0	37		or
0	RS	RT	RD	0	38		xor
0	RS	RT	RD	0	39		nor
0	RS	RT	RD	0	42		slt
0	RS	RT	RD	0	43		sltu
1	RS	0	offset				bltz
1	RS	1	offset				bgez
1	RS	16	offset				bltzal
1	RS	17	offset				bgezal
2	Adresse mot						j
3	Adresse mot						jal
4	RS	RT	offset				beq
5	RS	RT	offset				bne
6	RS	0	offset				blez
7	RS	0	offset				bgtz

8	RS	RD	Constante signée		addi
9	RS	RD	Constante signée		addiu
10	RS	RD	Constante signée		slti
11	RS	RD	Constante signée		sltiu
12	RS	RD	Cte non signée		andi
13	RS	RD	Cte non signée		ori
14	RS	RD	Cte non signée		xori
15		RD	Cte non signée		lui
16	0	RD	CS		mfco
16	4	RS	CD		mtco
16	8	0	offset		bcof
16	8	1	offset		bcot
16	16	0		1	tlbr
16	16	0		2	tlbwi
16	16	0		6	tlbwr
16	16	0		8	tlbp
16	16	0		16	rfe
32	RS	RD	offset		lb
33	RS	RD	offset		lh
35	RS	RD	offset		lw
36	RS	RD	offset		lbu
37	RS	RD	offset		lhu
40	RS	RT	offset		sb
41	RS	RT	offset		sh
43	RS	RT	offset		sw



Registres	Nom
\$0	\$zero
\$1	\$at
\$2	\$v0
\$3	\$v1
\$4,\$5,\$6,\$7	\$a0 ... \$a3
\$8, ..., \$15	\$t0 ... \$t7
\$16, ..., \$23	\$s0 ... \$s7
\$24, \$25	\$t8, \$t9
\$26, \$27	\$k0, \$k1
\$28	\$gp
\$29	\$sp
\$30	\$s8
\$31	\$ra



	000	001	010	011	100	101	110	111
000	<b>SPECIAL</b>	<b>BCOND</b>	J	JAL	BEQ	BNE	BLEZ	BGTZ
001	ADDI	ADDIU	SLTI	SLTIU	ANDI	ORI	XORI	LUI
010	<b>COPRO</b>							
011								
100	LB	LH		LW	LBU	LHU		
101	SB	SH		SW				
110								
111								

OPCODE = SPECIAL (INS 5:0)

	000	001	010	011	100	101	110	111
000	SLL		SRL	SRA	SLLV		SRLV	SRAV
001	JR	JALR			SYSCALL	BREAK		
010	MFHI	MTHI	MFLO	MTLO				
011	MULT	MULTU	DIV	DIVU				
100	ADD	ADDU	SUB	SUBU	AND	OR	XOR	NOR
101			SLT	SLTU				
110								
111								