

COGNOMS _____ NOM _____

Problema 1. (1 punt)

Donats els fragments de codi següents, en llenguatge màquina, escriviu en cada cas el fragment de codi equivalent en llenguatge C. Cas que sigui necessari podeu definir noms de variables i el seu tipus més adient.

<u>Instruccions</u>	<u>Comentari</u>	
mov R0, 1000	;Adreçament De registre directe i Directe	1000 <- X 1001 <- Y
mov R1, 1001		if (X!= Y)
cmp R0, R1	;Comparació	X = X * 2;
je <label1>	;Salta si igual	else
shl R0	;Desplaçament a l'esquerra	X = X + Y;
mov 1000, R0		Y = Y - X
jmp <label2>	;Salt incondicional	
label1: add R0, R1	;Suma	
mov 1000, R0		
label2: sub R1, R0	;Resta	
mov 1001, R1		

Problema 2. (1 punt)

Donat el fragment de codi següent, en llenguatge C, escriviu el fragment de codi equivalent en llenguatge màquina. Cas que sigui necessari podeu assignar a cada variable l'adreça que considereu adient.

```
#define N 1000
```

```
int i, max, V[N];
```

```
...
```

```
max = V[0];
```

```
for ( i = 1; i < N; i++ )
```

```
    if ( max < V[i] ) max = V[i];
```

```
i <- 999    max <- 998    v <- 1000
```

```
    mov R0,v
    mov X, #1
for:    cmp X, #1000
```

```
jge fifor
cmp R0, v[X]
jge fiif
mov R0, v[X]
fiif:    inc X
        jmp for
fifor:    mov 998, R0
        mov 999, X
```

Problema 3. (1 punt)

Quin és el valor en decimal (base 10) de la seqüència de bits següent 11010110 si la interpretem com:

- a) Un número natural en binari (0,25 punts)

$$11010110 = 2^7 + 2^6 + 2^4 + 2^2 + 2^1 = 128 + 64 + 16 + 4 + 2 = 214_{(10)}$$

- b) Un número enter representat en signe magnitud i 8 bits (SM) (0,25 punts)

$$11010110 = - (2^6 + 2^4 + 2^2 + 2^1) = - (64 + 16 + 4 + 2) = -86_{(10)}$$

- c) Un número enter representat en complement a 2 i 8 bits (Ca2) (0,5 punts)

$$11010110 = - (00101010) = - (2^5 + 2^3 + 2^1) = - (32 + 8 + 2) = -42_{(10)}$$

Problema 4. (1 punt)

Quin és el valor en binari (base 2) de la seqüència de dígit 321 si la interpretem com:

- a) Un número natural en base 16 (és a dir, $321_{(16)}$) (0,25 punts)

$$001100100001$$

- b) Un número natural en base 8 (és a dir, $321_{(8)}$) (0,25 punts)

$$011010001$$

- c) Un número natural en base 10 (és a dir, $321_{(10)}$) (0,5 punts)

$$321 \text{ div } 2 = 160, \text{ Residu} = 1$$

$$160 \text{ div } 2 = 80, \text{ Residu} = 0$$

$$80 \text{ div } 2 = 40, \text{ Residu} = 0$$

$$40 \text{ div } 2 = 20, \text{ Residu } 0 \quad . . .$$

$$101000001$$

Problema 5. (1 punt)

Donats els valors A = 11110101 i B = 11100001 que representen dos números enters binaris en complement a 2 (Ca2) i 8 bits. (Opereu sempre en Ca2)

- a) Quin és el valor de A + B? Es produeix desbordament (expliqueu el perquè)?

```
1 1 1 1 0 1 0 1
1 1 1 0 0 0 0 1
1 1 0 1 0 1 1 0  C=1  Overflow = 0 perquè el signe dels operands és igual al del resultat
```

- b) Quin és el valor de A - B? Es produeix desbordament (expliqueu el perquè)?

```
-B = 00011111

1 1 1 1 0 1 0 1
0 0 0 1 1 1 1 1
0 0 0 1 0 1 0 0  C=1  Overflow = 0 perquè estem sumant un positiu i un negatiu.
```

Problema 6. (1 punt)

Donats els valors A = 11110101 i B = 11100001 que representen dos números enters binaris en complement a 1 (Ca1) i 8 bits. (Opereu sempre en Ca1)

- a) Quin és el valor de A + B? Es produeix desbordament (expliqueu el perquè)?

```
1 1 1 1 0 1 0 1
1 1 1 0 0 0 0 1
1 1 0 1 0 1 1 0
_____1 C=1
1 1 0 1 0 1 1 1  Overflow = 0 perquè el signe dels operands és igual al del resultat
```

- b) Quin és el valor de A - B? Es produeix desbordament (expliqueu el perquè)?

```
-B = 0 0 0 1 1 1 0

1 1 1 1 0 1 0 1
0 0 0 1 1 1 1 0

0 0 0 1 0 0 1 1
_____1 C=1
0 0 0 1 0 1 0 0  Overflow = 0 perquè estem sumant un positiu i un negatiu.
```

Problema 7. (2,5 punts)

Donat el següent fragment de programa

1.-	mov X, #0	50.-	push R0
2.-	mov R0, 200[X]	51.-	inc R0
3.-	mov (301), R0	52.-	inc X
4.-	call 50	53.-	pop R0
5.-	cmp X, #2	54.-	ret
6.-	jnz 2		

Mostreu l'evolució dels registres, la memòria i la pila.

INSTRUCCIO	PC	SP	R0	X	100	101	102	200	201	202	300	301	302	400	401
	1	400			200	201	202	10	11	12	100	101	102		
mov X, #0	2			0											
mov R0, 200[X]	3		10												
mov (301), R0	4					10									
call 50	50	401												5	
push R0	51	402													10
inc R0	52		11												
inc X	53			1											
pop R0	54	401	10												
ret	5	400													
cmp X, #2	6														
jnz 2	2														
mov R0, 200[X]	3		11												
mov (301), R0	4					11									
call 50	50	401												5	
push R0	51	402													11
inc R0	52		12												
inc X	53			2											
pop R0	54	401	11												
ret	5	400													
cmp X, #2	6														
jnz 2	7														
...															

Els modes d'adreçament utilitzats són:

n Directe #n immediat n[X] Indexat (n) Indirecte

L'SP apunta a la 1a posició lliura de la pila i aquesta creix cap a posicions majors de memòria.

