

# EC Examen de Problemes

## Exercici 1 (examen final 12/13 Q1)

Donada la següent declaració de variables globals, que s'ubica a memòria a partir de l'adreça 0x10010000:

```
.data
v1: .byte 12
v2: .half 0x13F
v3: .dword -4
v4: .ascii "01234" # codi ASCII '0' = 0x30
v5: .byte 3
v6: .space 5
v7: .float 3.5
```

Ompla la següent taula amb el contingut de memòria en hexadecimal. Les variables s'emmagatzemen a partir de l'adreça 0x10010000. Les posicions de memòria sense inicialitzar es deixen en blanc.

@Memòria	Dada	@Memòria	Dada	@Memòria	Dada	@Memòria	Dada
0x10010000		0x10010010		0x10010020		0x10010030	
0x10010001		0x10010011		0x10010021		0x10010031	
0x10010002		0x10010012		0x10010022		0x10010032	
0x10010003		0x10010013		0x10010023		0x10010033	
0x10010004		0x10010014		0x10010024		0x10010034	
0x10010005		0x10010015		0x10010025		0x10010035	
0x10010006		0x10010016		0x10010026		0x10010036	
0x10010007		0x10010017		0x10010027		0x10010037	
0x10010008		0x10010018		0x10010028		0x10010038	
0x10010009		0x10010019		0x10010029		0x10010039	
0x1001000A		0x1001001A		0x1001002A		0x1001003A	
0x1001000B		0x1001001B		0x1001002B		0x1001003B	
0x1001000C		0x1001001C		0x1001002C		0x1001003C	
0x1001000D		0x1001001D		0x1001002D		0x1001003D	
0x1001000E		0x1001001E		0x1001002E		0x1001003E	
0x1001000F		0x1001001F		0x1001002F		0x1001003F	

## Exercici 2 (problema 2.29 de la col·lecció)

Donades les següents declaracions de variables globals, en llenguatge C:

```
char *punterc;
short *punterh;
int *punteri;
long long *punterd;
```

Tradueix a ensamblador MIPS les següents sentències:

- a) `punterc++;`
- b) `punteri++;`
- c) `punterh++;`
- d) `punterd++;`
- e) `*punteri = *punteri + 5;`
- f) `*punterh = *punterh + 10;`

### Exercici 3 (problema 2.28 de la col·lecció)

Donades les següents declaracions de variables globals, en C:

```
int vec[6] = {2, 4, 0, 6, 8, 0};  
int *punter;
```

Tradueix a una única sentència en C el conjunt d'instruccions de cada apartat:

- a) 

```
la $t0, punter  
la $t1, vec + 8  
sw $t1, 0($t0)
```
- b) 

```
la $t0, punter  
lw $t1, 0($t0)  
addiu $t1, $t1, 4  
sw $t1, 0($t0)
```
- c) 

```
la $t1, vec  
lw $t3, 0($t1)  
lw $t4, 4($t1)  
addiu $t3, $t3, $t4  
sw $t3, 4($t1)
```
- d) 

```
la $t1, vec  
la $t0, punter  
lw $t2, 0($t0)  
lw $t3, 0($t2)  
addiu $t3, $t3, 1  
sw $t3, 8($t1)
```
- e) 

```
la $t0, punter  
lw $t2, 0($t0)  
lw $t3, 0($t2)  
addiu $t3, $t3, 1  
sw $t3, 8($t2)
```