



## Final de PBN

Enginyeria de Sistemes TIC

105 MINUTS

27 de juny de 2013

COGNOMS:

NOM:

GRUP de LAB:

**Exercici 1 [1 punt].** Considereu les dues capçaleres següents:

```
int funcio_a();  
int funcio_a(void);
```

Quina de les següents sentències és certa?

- Ambdues capçaleres són equivalents.
- La primera capçalera és errònia.
- La segona capçalera és errònia.
- La funció `int funcio_a(int k){return k;}` és compatible amb la primera capçalera però no amb la segona.

**Exercici 2 [1 punt].** Considereu el següent codi C99:

```
int8_t j=0;  
for (int i=0; i<10;) j+=--i;
```

Quina de les següents afirmacions és correcta?

- La iteració no acaba mai.
- El codi té errors sintàctics.
- En acabar el codi, la variable `j` val -45.
- A la variable `j`, que és de tipus `int8_t` no se li pot assignar un valor de tipus `int`.

**Exercici 3 [1 punt].** Quina de les següents afirmacions sobre la taula de símbols és correcta:

- La taula de símbols conté els identificadors de totes les variables globals i locals del mòdul.
- La taula de símbols conté els identificadors de totes les funcions del mòdul.
- La taula de símbols conté les adreces del mòdul que cal reubicar.
- La taula de símbols conté els timestamps del mòdul.

**Exercici 4 [1 punt].** Considereu el següent fragment de codi C:

```
typedef struct {uint8_t a,b;} t1[10];  
t1 a,*b;
```

Quina de les següents expressions és correcta?

- a = b
- \*(a+1).a = b[2] -> a
- &a == b
- (\*b).a = 4

**Exercici 5 [1 punt].** Supposeu que un `Makefile` conté les següents dependències:

```
a.o: b.h c.h d.h  
b.o: d.h  
a: c.o b.o d.o
```

Assumint les regles per defecte, quina de les següents afirmacions és certa?

- La tercera regla del `Makefile` és errònia.
- Si recompillem `a.c` també cal recompilar `b.c`.
- Si modifiquem `c.h` caldrà recompilar tots els mòduls.
- Per forçar una recompilació de tots els mòduls podem modificar `d.h` i `c.h`.

**Exercici 6 [1 punt].** Supposeu que els mòduls `a.c` i `b.c` d'un mateix programa contenen tots dos el següent codi:

```
#include ...  
  
uint8_t v2 = 0;
```

Argumenta quines implicacions tindrà sobre l'executable.

**Exercici 7 [2 punts].** En Pau es mira el següent fragment de codi del mòdul `ether` de la pràctica:


```

void ether_put(uint8_t c) {
    mchar_t mc;
    ...
    while (queue_is_full(&outq));
    queue_enqueue(&outq, mc);
    ...
}

```

Després d'analitzar-lo diu:

— El codi no sembla correcte: segons on es produeixi una interrupció que afegixi un element a `outq`, `enqueue` fallaria. Argumenta sobre la veracitat o no d'aquesta afirmació assumint que `queue_is_full()` i `queue_enqueue()` estan ben implementades.

**Exercici 8 [2 punts].** Situeu-vos en el context de les pràctiques. Imagineu que volem transmetre el senyal morse de forma diferent: tant punts com ratlles tenen la mateixa durada (un DOT exactament) però el punt es representa amb un so de 3.1 kHz i la ratlla amb un silenci. L'enviament d'un caràcter morse va precedir sempre d'un punt que s'usa per sincronitzar el receptor. Per exemple, per enviar el caràcter R, punt-ratlla-punt, el senyal que cal enviar és  (2 quadres = 1 DOT).

Feu un redisseny de l'emissor del mòdul `ether` del projecte que incorpori aquesta modificació.

**Exercici 9 [5 punts].** L'objectiu d'aquest exercici és dissenyar i implementar un mòdul que ofereixi un controlador d'una porta automàtica basant-se en alguns del mòduls que heu definit a les pràctiques.

La porta automàtica es comanda amb un polsador d'apertura. Quan es dona l'ordre d'obrir comença el cicle d'apertura excepte que la porta ja estigui oberta o bé estigui obrint-se. Una vegada oberta, la porta roman així indefinidament.

En el cicle d'apertura cal engegar el motor d'apertura fins que el sensor de porta oberta indica que la porta ja està oberta. Si el sensor d'apertura no dona senyal 60s després d'haver engegat el motor és que la porta s'ha encallat. En aquest cas cal engegar el llum d'alarma durant 200s. El sensor d'alarma encén intermitentment el llum d'alarma amb un període de 1s. Si es produeix una alarma les noves senyals d'apertura de porta s'ignoren indefinidament.

Es demana que, assumint que teniu disponibles tots els mòduls del projecte, dissenyeu i implementeu aquest mòdul. Per implementar-lo baseu-vos en els serveis del mòdul `timer`, del qual podeu assumir la resolució que us convingui. També cal que assumiu que teniu les següents funcions disponibles:

```
/*
 * Si accio = 0 -> para el motor
 * Si accio = 1 -> engega el motor
 */
void motor_porta(uint8_t accio);

/*
 * Si accio = 0 -> para el llum
 * Si accio = 1 -> engega el llum
 */
void llum_emer(uint8_t accio);
```

A més, el sensor de porta oberta i el del polsador d'obrir porta actuen per interrupcions. Les rutines de servei d'interrupcions corresponents són aquestes:

```
void isr_porta_oberta(void);
void isr_polsador_porta(void);
```

Es valorarà la capacitat d'usar els mòduls disponibles de manera eficaç, la bondat de l'estructura del programa resultant i la claredat en el desenvolupament de la resposta.