

**Control de Laboratori**  
**Grup 30**  
**Contesteu al mateix full**

---

Nom i Cognoms:

---

1. (3 punts) Seleccioneu la resposta ( una només ) que considereu correcta en cadascún dels apartats. Cada resposta correcta val 0,5 punts. Cada resposta incorrecta resta 0,25.
- (a) Indica quina de les següents afirmacions es certa:
- Salvar posa els parametres d'usuari a la pila
  - El procés nul ha de tenir el pid 0.
  - Cada PCB guarda el quantum que li queda al procés corresponent.
  - La rutina primer retorna un punter a PCB.
- (b) En crear un procés, el valor inicial del registre AX ha de ser:
- Un 0 per indicar que la creació s'ha realitzat correctament.
  - Un 0 per inicialitzar-lo a un valor neutre.
  - El pid del procés.
  - Cap de les respostes anteriors.
- (c) A la inicialització del sistema s'ha de:
- Inicialitzar el PCB del procés nul
  - Inicialitzar el punter de READY
  - Crear el primer procés d'usuari
  - Totes les anteriors.
- (d) A crear\_pcb\_nuc s'ha de:
- Posar en execució el nou procés.
  - Retornar el pid del nou procés.
  - Buscar una pila per el nou procés
  - Totes les anteriors.
- (e) Per inicialitzar la taula de PCBs hem de:
- Reservar memòria per els PCBs.
  - Marcar cada PCB com a lliure.
  - Posar a 0 el temps de cpu de cada PCB.
  - Totes les anteriors.
- (f) Indica quina de les següents afirmacions es falsa:
- El procés nul no té una pila assignada per el sistema.
  - El procés nul no té codi ja que no fa res.
  - El procés nul té la prioritat mínima del sistema.
  - El procés nul te el quantum mínim.
2. (2 punts) Explica **raonadament** el context de qui es restaura a la rutina salvar i perquè es necessari. A on es salva aquest context?

3. (2 punts) Explica **raonadament** perquè hem de crear un context artificial quan creem un nou procés.

4. (3 punts) Troba 3 errors a la següent rutina multiplexar corresponent a una pràctica d'Onion. Indica en cada cas que treuries, afagirias o modificarias raonant la resposta. Els errors no son de caràcter sintàctic. Un únic error pot implicar diversos canvis.

```
void multiplexar ( )
{
    PCB *aux;
    RUN->ctx = salvar();
    if ( !buida(&qReady) ) {
        aux = (struct PCB *)primer(&qReady);
        if ( aux->prioritat > RUN->prioritat ) {
            insertar(&qReady,(struct item *)RUN,RUN->prioritat);
            RUN = aux;
            tics = RUN->quantum;
        } else {
            if ( --tics == 0 && aux->prioritat == RUN->prioritat ) {
                insertar(&qReady,(struct item *)RUN,RUN->prioritat);
                RUN = aux;
                tics = RUN->quantum;
            }
        }
    } else {
        if ( --tics == 0 ) tics = RUN->quantum;
    }
    restaurar(RUN->ctx);
}
```

(a)

(b)

(c)