



Izen-deiturak:	Taldea: Euskara Taldea
-----------------------	----------------------------------

Estimatutako azterketaren iraupena: 2,5 ordu

1º [2]	2º [3]	3º [1,5]	4º [1]	5º [2,5]		
-----------	-----------	-------------	-----------	-------------	--	--

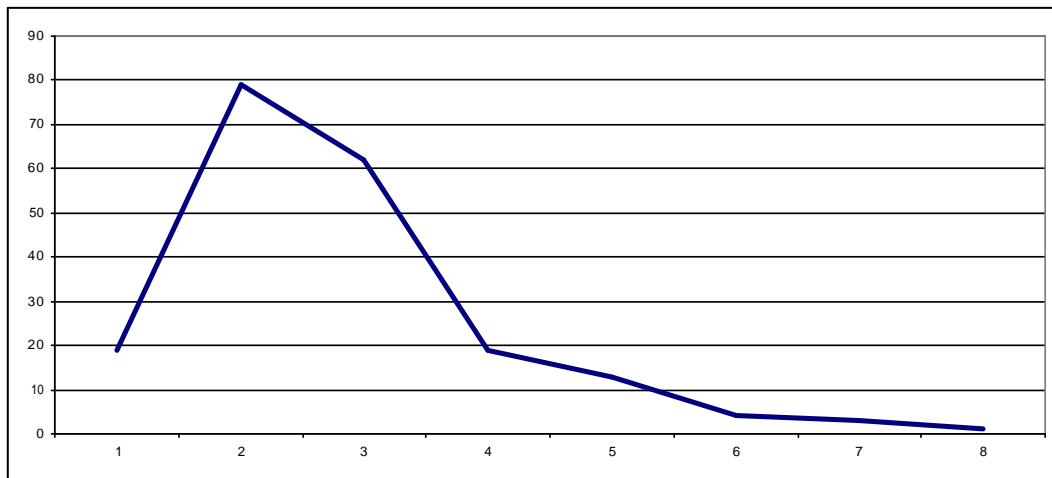
1 Ariketa [2 puntu]

Erantzun itzazu ondoko galderak gehienez 3 lerro erabiliz (eta noski letra txikiegia erabili gabe).
Lerro gehiagoren erabilerak puntuen galtzea suposatuko du.

1. Azaldu modu laburrean zertan desberdintzen diren ondoko kontzeptuak: *programa*, *prozesua* eta *thread*
2. Elkarblokeaketaren arazoari aurre egiteko soluzio posibleen artean *predikzioa* eta *prebentzioa* aurki ditzakegu. Zertan desberdintzen dira bi hauek?

3. Sistemaren errendimendua neurtzeko asmoz, gure prozesuen CPU tarteen tamaina kalkulatzeko esperimentu bat egin dugu. Azterketa honetan ordenadore bakar bat erabili da, eta erabiltzaileek exekutatzen dituzten aplikazio arruntenak aukeratu ditugu. Jasotako emaitzak ondoko taula eta grafikoan jaso ditugu:

CPU tartea (tick)	Maiztasuna (zenbatetan aurkitua)		Metatutako maiztasuna	
	absolutua	Erlatiboa	Absolutua	erlatiboa
1	19	% 9,5	19	% 9,5
2	79	% 39,5	98	% 49
3	62	% 31	160	% 80
4	19	% 9,5	179	% 89,5
5	13	% 6,5	192	% 96
6	4	%2	196	% 98
7	3	%1,5	199	% 99,5
8	1	% 0,5	200	% 100



Gure sistemaren quantum-aren balioa une honetan 2 tick-ekoa da. Gure esperimentuaren emaitzak kontuan izanik, erantzun itzazu ondoko galderak:

- 3.a. Quantum-aren balioa egokia dela uste al duzu? Horrela ez balitz, zenbatean jarriko zenuke?

2 Ariketa [3 puntu]

Demagun sistema eragile batean ondoko prozesuen informazioa dugula:

<u>Prozesua</u>	<u>Lehentasun-maila</u>
P1	5
P2	6
P3	3
P4	5

Non prozesuek exekutatzen dituzten programen kodea ondokoa den:

P1

```
{
    while (1) {
        (5 CPU tick)
        esnatu(P3);
        (3 tick blokeatuta)
        esnatu(P4);
    }
}
```

P2

```
{
    while (1) {
        (2 CPU tick)
        (11 tick blokeatuta)
    }
}
```

P3

```
{
    while (1) {
        (5 CPU tick)
        lotara();
    }
}
```

P4

```
{
    while (1) {
        (2 CPU tick)
        lotara();
    }
}
```

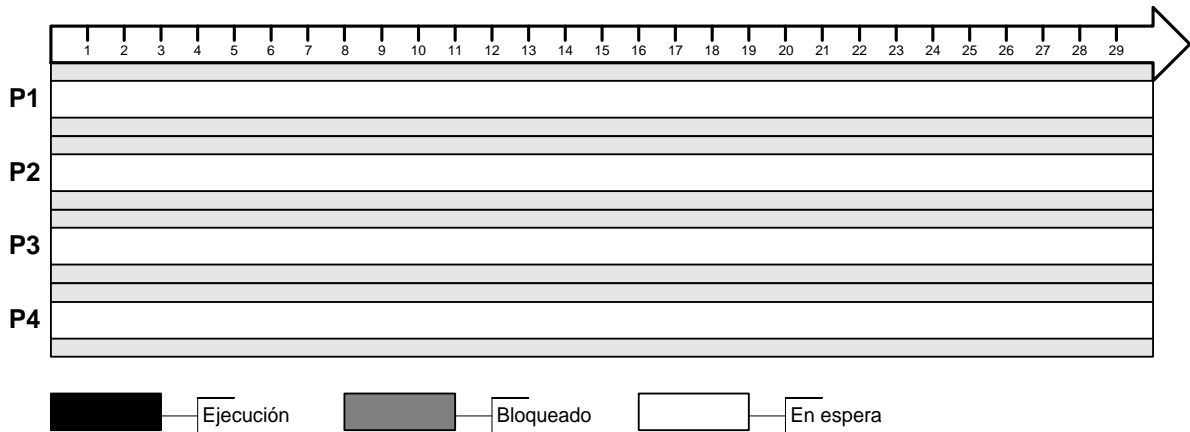
Horrela, **P1** prozesuak amaiera gabeko begizta bat exekutatzen du, non ziklo bakoitzean 5 CPU-tick exekutatu ondoren P3 prozesua esnatzeko aginduari deitu eta 3 tick-eko iraupena duen S/Iko eragiketa bat egikaritzen du; eragiketa burutu ondoren, P4 prozesua esnatzeko deia egin eta hurrengo begizta exekutatzerara pasatzen da. Programa guztiek amaiera gabeko begiztak dituzte.

Kontuan izanik:

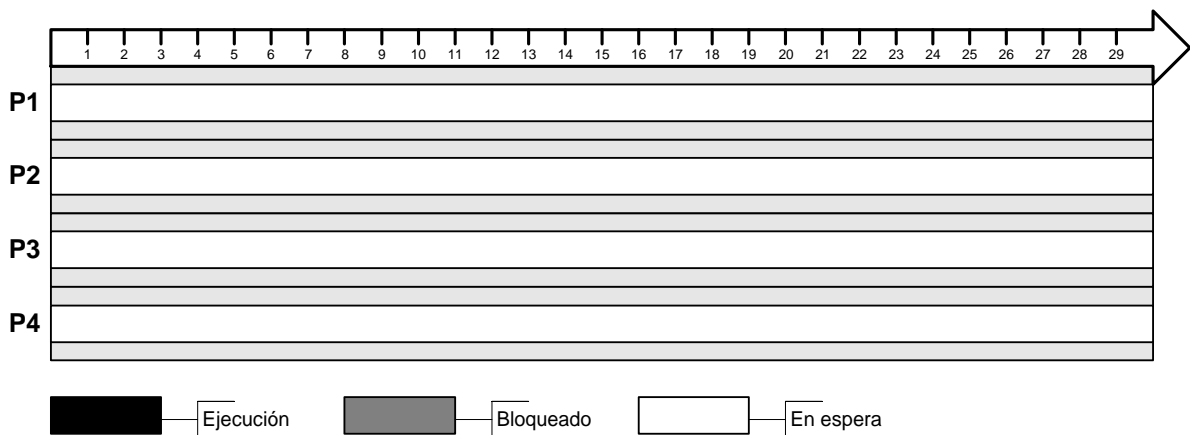
- lehentasun-maila handiagoak lehentasun handiagoa adierazten duela
- **lotara** eta **esnatu** eragiketen exekuzio-denborak oso txikiak direnez, arbuia egingo ditugula.

2.a. Elkarblokeaketa arazorik gerta al daitezke prozesu hauek konkurrentzian exekutatzen badira? Arrazoitu zure erantzuna.

2.b. Marraz ezazu ondoko exekuzio-kronogramaren lehendabiziko 30 tick-en egoera, ondoko planifikazio politikarako: **FCFS eta Round Robin**, quantum balioa 3 tick-ekoa izanik. Suposatu lau prozesuak 0.tick-ean hasi direla denak elkarrekin eta "P1, P2, P3 eta P4" ordenean iritsi direla.



2.c. Marraz ezazu ondoko exekuzio-kronogramaren lehendabiziko 30 tick-en egoera, ondoko planifikazio politikarako: **lehentasun estatikoak eta Round Robin**, quantum balioa berriro ere 3 tick-ekoa izanik. Suposatu ere lau prozesuak 0.tick-ean hasi direla denak elkarrekin eta "P1, P2, P3 eta P4" ordenean iritsi direla. Suposa ezazu politika hau gertaera eta denbora bidezko kanporatzailea dela.



2.d. Kalkula ezazu sistemaren CPU-tasa proposatutako bi politikentzako (2.b. eta 2.c. atalak).

2.e. Kalkula ezazu bi politikentzako (2.b. eta 2.c.) batezbesteko erantzun-denborak (t_r).

2.f. Aurreko ataletan egindako kronogramen eta kalkulaturako errendimenduaren neurtzeko parametroen arabera, (2.b) eta (2.c) ataletako aukeren artean, zein deritzozu dela planifikazio-politika egokiena zure sisteman ezartzeko? Zergatik? (ez duzu ezer kalkulatu behar, arrazoitu erantzuna soilik)

2.g. Bigarren kronogramari begira (c.), kalkula ezazu eraginkortasunaren galera ($1 - E_f$) testuinguru aldaketak direla eta, kontuan izanik 1.400 MHz-etako prozesadore bat dugula, sistemaren erlojuaren maiztasuna 100 Hz dela, eta testuinguru-aldaketa bakoitzaren denbora (t_{cs}) 1 ms dela.

3 Ariketa [1,5 puntu]

Demagun ordenadore batean 64 Mbyteko memoria fisikoa dugula, eta bertan ezarri den sistema eragileak memoria segmentatua erabiltzen duela, helbide logikoek ondoko egitura izanik: 10 bit segmentu atalerako eta 22 bit desplazamendurako. Une konkretu batean, memoriaren egoera ondokoa da: sistema eragilea memoria fisikoaren hasierako 128 Kbytetan dago kokatua, eta **P** prozesu bat dago martxan hori segmentu esleituta dituelarik, bakoitzaren tamaina 10 KByte, 20 KByte eta 2 KByte izanik, eta ondoko helbide fisikoetan kokatu direlarik: 300.000, 200.000 eta 700.000.

3.a. Marraz ezazu memoria kudeaketa sistema honen helbide itzulpenaren eskema, helbide logiko eta fisikoen osagaiak eta tamainak adieraziz.

3.b. Marraz ezazu **P** prozesuaren segmentu-taula, argi adieraziz bere edukia atal (gelaxka) bakoitzean.

3.c. Zein helbide fisikoa dagokio 2. segmentuan 1000 desplazamendua duen helbide logikoarekin?

3.d. Sistemak *best-fit* asignazio-politika erabiltzen badu, eta 70 KByteko segmentu bat memorian kokatzeko eskaera jasotzen badu, zein memoria tartea aukeratuko litzateke segmentua gordetzeko?

4 Ariketa [puntu 1]

Memoria orrikatua duen kudeaketa sistema batean, programa batek 5 orri ditu guztira (0-tik 4-ra bitarte zenbakituak). Orriaren tamaina 4Kbytekoa da. Une konkretu batean, exekuzioan zehar, ondoko orri-
taula dugu:

markoa	Baliagarritasun bita
3	1
4	1
10	0
2	1
1	0

Prozesuak ondoko helbide logikoak atzitu nahi ditu: 2500, 5300, 9600 eta 19500 (hamartarrez)

4.a. Orri hutsegiterik ematen al da? Baiezko kasuan, adierazi zein kasu diren eta arrazoitu erantzuna.

4.b. Adieraz ezazu zein helbide fisikoei dagozkien orri-hutsegiterik sortzen ez duten helbide logikoak.

5 Ariketa [2,5 puntu]

Disko batean gordeta dugun fitxategi baten ondoko kontrol-informazioa dugu:

- Fitxategiaren izena: `datuak.dat`
- Tamaina (bloketan): 15
- Datu-blokeak diskoan (bloke absolutuak): 5,6,11,8,15,12,4,13,16,7,2,3,9,17,14.

Diskoak ondoko geometria du: 1024 zilindro, 8 pista/zilindro, 32 sektore/pista, eta 512 bytetako sektoreak. Honez gain, badakigu disko kontroladoreak sektore bat DMAz transferitzeko batez beste 7 μ s behar dituela, diskoaren errotazio-abiadura 6.000 bira/minutukoa dela, eta burua kokatzeko batez besteko denbora 11 ms-koa dela. Kontuan izanik diskoaren 0 eta 1 sektoreak erreserbatuta daudela, sektore kalteturik ez dagoela, eta diskoak gordetzen duen fitxategi bakarra `datuak.dat` fitxategia dela, erantzun itzazu ondoko galderak:

5.a. FAT16 fitxategi-sistema batekin lan egiten badugu, adieraz ezazu zein izango den katalogoaren eta FAT-aren edukia.

5.b. Aldiz, UNIXen fitxategi-sistemarekin lan egingo bagenu, adieraz ezazu beste kasu honetarako zein izango litzateke katalogoaren edukia eta `datuak.dat` fitxategia kudeatzeko erabiliko liratekeen datu-egiturak. Zalantzako kasuan, egin itzazu egokien deritzozun suposaketak.

5.c. Demagun gure fitxategi-sistemaren blokea 2 Kbytekoa dela. Fitxategia 256 bytetako erregistro logikotan antolatuta dago, eta diskoaren sektorearen tamaina (bloke fisikoa) 1 Kbytekoa da. Aplikazio batek 52. erregistro logikoa atzitu nahi badu, zein izango da helbideen itzulpen prozesua? Adieraz ezazu zeintzuk izango diren bloke erlatibo, bloke absolutu eta zilindro-pista-sektore balioak¹.

5.d. Aurreko kasurako, zenbat atzipen fisiko egin beharko ditugu FAT 16 eta UNIX fitxategi-sistema duten bi diskoetan? Suposa ezazu katalogoak, FAT taula eta i-nodoen informazioa memorian kargatuta daudela.

5.e. Kontuan izanik disko honen kontroladoreak 4 faktoreko hardware interleaving-a aplikatzen duela, kalkula ezazu bloke logiko bat atzitzeko beharko den batez besteko denbora.

¹ Horretarako ondoko eragiketak erabil ditzakezu behar izanez gero: **sektorea**= $b\%S$; **zilindroa**= $b \operatorname{div} (S*P)$; **pista** = $(b\%(S*P)) \operatorname{div} S$; non S= sektore kopurua, P=pista kopurua, eta b=bloke zenbakia