

Oinarrizko Programazioa – Hirugarren partziala.

2020ko urtarilak 10 – Ebaluazioa Jarraia

Irakasgaia gainditzeko, azterketa honetan 2 puntu atera behar dira, nota osoaren % 40a. Gainera, hiru partzialen batezbesteko ponderatua, hamarretik 4.5 puntu atera beharko dira.

1. Saskibaloia partidua – ADA (3 puntu)

Saskibaloia partidu batean egiten diren jaurtiketa guztiak, sartutakoak eta sartu ez direnak, gordetzeko ondoko datu egitura dugu.

```
Max_Jaurtiketa: constant Integer := 10000;

type T_Talde is (Etxeko, Kanpoko);

type T_Jokalari is record
    Talde: T_Talde;
    Izen: String(1..50);
    Dortsal: Natural;
end record;

subtype T_Balio is Integer range 1..3;    -- 1, 2 edo 3 puntu

type Jaurtiketa is record
    Jokalari: T_Jokalari;
    Balio: T_Balio;      -- jaurtiketaren balioa(batekoa, biko edo hirukoa)
    Sartu: Boolean;      -- True sartu bada, False ez bada sartu
end record;

type Jaurtiketen_Bektore is array(1.. Max_Jaurtiketa) of Jaurtiketa;

type Jaurtiketen_Zerrenda is record
    Zenbat: Integer;
    Jaurtiketak: Jaurtiketen_Bektore; -- Ez du inongo orden zehatzik
end record;
```

- A. (0.5 puntu) Ondoko azpiprograma inplementatzea, non jaurtiketan zerrenda eta jokalaria bat emanda, jokalaria horrek hiruko jaurtiketarik sartu duen adierazten duen Boolean bat itzuliko duen:

```
function hiruko_sartu_du (L: in Jaurtiketen_Zerrenda; J: in T_Jokalari)
    return Boolean
    -- aurre: L partiduan egindako jaurtiketa guztiak ditu eta J
    --          jokalaria baten informazioa.
    -- post: True itzuliko du J-ak partiduan gutxienez hiruko bat sartu
    --          badu (3 puntutako jaurtiketa), eta False bestela
```

Oharra: Atal hau zuzentzerakoan eraginkortasuna bereziki hartuko da kontutan.

B1. (puntu 1) B2 atala egiteko beharko den *Jaurtiketen_Hedapena* datu egitura definitzea.

Oharra: Talde bakoitzak gehienez 12 jokalaria izan ditzake.

B2. (1.5 puntu) *Jaurtiketen_Zerrenda* bat jaso eta *Jaurtiketen_Hedapena* datu-egitura osatzen duen prozedura zehaztea eta inplementatzea. *Jaurtiketen_Hedapena* datu-egitura talde bakoitzeko jaurtiketaren bat egin duten jokalarien zerrenda gordeko du (dorsal zenbakiaren arabera ordenatuta). Jokalari bakoitzeko, bere izena, dorsal zenbakia eta egindako puntu bateko, biko zein hiruko jaurtiketan kontaktak (sartutakoak bakarrik, jaurtitako guztiak eta sartutakoen ehunekoak jaurtitako guztien artean) gordeko dira, ondoko adibidean erakusten den moduan.

ETXEKO TALDEA										
Dorsala	Izena	T1 Sartu	T1 Jaurti	T1%	T2 Sartu	T2 Jaurti	T2%	T3 Sartu	T3 Jaurti	T3%
2	P. Paz	4	4	100%	0	0	0%	0	1	0%
11	Severiano	0	1	0%	1	3	33%	2	5	40%
12	Da Silva	2	3	66%	2	2	100%	0	0	0%
...
KANPOKO TALDEA										
Dorsala	Izena	T1 Sartu	T1 Jaurti	T1%	T2 Sartu	T2 Jaurti	T2%	T3 Sartu	T3 Jaurti	T3%
7	R. Will	5	5	100%	4	5	80%	3	3	100%
11	Asensio	0	0	0%	0	1	0%	0	2	0%
13	Urkiola	0	0	0%	3	9	33%	1	2	50%
...

OHARRA: Atal hau ebazteko, gutxienez, ondoko azpiprogramak idaztea eta erabiltzea gomendatzen da (eta baloratuko da):

- *dorsalaren_posizioa* (*Hed*: in *Jaurtiketen_Hedapena*; *Talde*: in *T_Talde*; *Dorsal*: in *Natural*; *Dago*: out Boolean; *Pos*: out Integer). Azpiprograma honek jaurtiketen hedapen bat (*Hed*), *Talde* bat eta *Dorsal* zenbaki bat hartzen ditu, eta dorsal hori hedapenean

dagoen ala ez adierazten du (*Dago*). Dortsala badago, *Pos* dortsal hori dagoen posizioa izango da, bestela, dortsal hori *egon beharko lukeen* posizioa izango da.

Nota: *Dago* aldagaia False izatea bilatzen den dortsala hedapenean ez dagoela esan nahi du. Beraz, momenturen batean elementu berri bat (dortsal) sartu beharko da hedapenean, ordena mantenduz. Hala ere, *Dortsalaren_posizioa* azpiprogramak, parametroak definitu diren moduagatik, ezin du txertaketa hori egin, eta jarraian deskribatzen den *Jokalaria_posizioan_sartu* azpiprogramak egingo du. *Dortsalaren_posizioa* azpiprogramak egin dezakena (eta horixe da ideia) topatzen ez den elementua *sartu beharko litzakeen* posizioa itzultzea da, ondoren *Jokalaria_posizioan_sartu* azpiprogramari parametro modura pasatzeko

- *jokalaria_posizioan_sartu* (.....). Azpiprograma honek jaurtiketan hedapenean, parametro modura jasotako taldean eta posizioan, jokalaria berri baten informazioa sartzen du (hau da, aurreko taulako lerro berri bat).
- *ehunekoak_kalkulatu* (.....). Azpiprograma honek jaurtiketen hedapen bat jasotzen du eta bi taldeen jaurtiketa ehunekoak (puntu batekoak, bikoak zein hirukoak) kalkulatu eta eguneratzen ditu. Horretarako jaurtiketa mota bakoitzeko hedapenean dagoen egindako eta sartutako jaurtiketen informazioa erabiliko du.

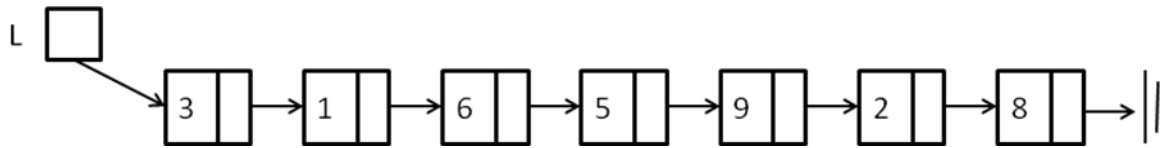
2. Elementuak ezabatu – ADA (1.25 puntu)

Ondoko zerrenda estekatua emanda:

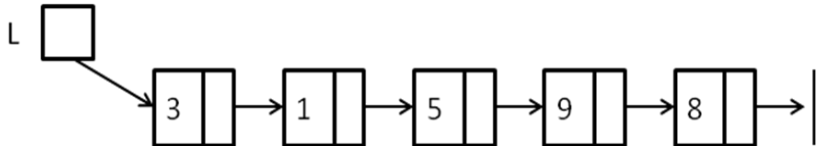
```
type Nodo;  
type Lista is access Nodo;  
type Nodo is record  
    Info: Integer;  
    Hurrengoa: Lista;  
end record;
```

Elementuak_ezabatu azpiprograma espezifikatu eta inplementatzea eskatzen da. Azpiprograma honek, zerrenda estekatu bat (L) eta zenbaki oso positibo bat (X) hartuko ditu eta L zerrendatik X, 2X, 3X ($nX \leq$ zerrendaren luzera den bitartean), eta abar posizioetan dauden elementuak ezabatuko ditu. Ezingo da beste zerrenda bat sortu, derrigorrez L zerrendaren gainean egin beharko da lan. Horretarako behar diren aldagai lagungarri guztiak erabiliz.

Zerrendako lehenengo elementua 1 posizioan dagoela kontsideratuko dugu. Hori dela eta, adibidez, L ondoko zerrenda balitz:



Elementuak_ezabatu(L, 3) deia exekutatu ondoren, L ondoko zerrenda adierazi beharko luke:



3. Karratu magikoa– PYTHON (0.75 puntu)

Karratu magiko bat zenbaki osokoen matrize karratu bat da, non matrizeko zutabeen, lerroen eta diagonal nagusien batura berdina den. Batura horri konstante magikoa deitzen zaio. Adibidez, ondoko matrizea karratu magiko bat adierazten du, edozein lerro, edozein zutabe eta bi diagonal nagusietako edozeinen batura berdina delako, hau da 34.

16	3	2	13
5	10	11	8
9	6	7	12
4	15	14	1

NxN tamainako osoko zenbakien matrize bat emanda, matrize hori karratu magiko bat adierazten duen ala ez adierazten duen azpiprograma zehaztea eta inplementatzea eskatzen da.

Programa eraginkorra izan behar du eta $N > 0$ edozein tamainarako funtzionatu behar du.