

Oinarrizko Programazioa – 1. partziala.

2022ko urriaren 28

Ariketa 1 – Simulazioa: 2 puntu

Laukian agertzen den **algoritmoaren simulazioa egitea** eskatzen da, laborategietan erabilitakoak bezalako taula baten bidez, erabiltzaileak teklatutako *zenb* balioa 3 den proba-kasurako.

```
zenb, kont, emaitza: Integer;
emaitza <-- 1;
kont <-- 0;
idatzi ("Sartu zenbaki oso bat: ");
irakurri(zenb);
bitartean (kont < 4) egin
    baldin (emaitza <= zenb) orduan
        zenb <-- zenb + zenb;
        emaitza <-- zenb;
    amaitu_baldin;
    kont <-- kont + 1;
amaitu_bitartean;
idatzi(emaitza);
```

Ariketa 2 – Harri-Orri-Ar (ADA): 5 puntu

Programa bat egin nahi dugu, erabiltzaileak harri-orri-ar-eko hainbat partida joka ditzan ordenagailuaren aurka. Partida bakoitzean, jokalaria bakoitzak (erabiltzaileak eta ordenagailuak) aukera bat aukeratzen du: harria, orria edo artazia; erabiltzaileak teklatu bidez aukeratzen du, eta ordenagailuak *jokaldi_random* azpiprograma erabiltzen du, aukeretako bat ausaz hautatzen duena. Horrela, jokaturako partida bakoitzean gerta daiteke erabiltzaileak irabaztea, ordenagailuak irabaztea edo bi jokalariek berdintzea, bakoitzak aukeratu duenaren arabera, eta honako hauek kontuan hartuta:

- Harriak artaziak apurtzen ditu (harriak irabazten du)
- Harriak harriarekin berdindu egiten du
- Artaziek papera mozten dute (artaziek irabazten dute)
- Artaziek artaziekin berdindu egiten dute
- Paperak harria estaltzen dute (paperak irabazten du)
- Paperak paperarekin berdindu egiten du

Honako (azpi)programa hauen **espezifikazioa eta Ada-zko inplementazioa** eskatzen dira:

- a) (0.75 puntu) **eskatu_gehienezko_partidak(...)** azpiprograma, erabiltzaileari eskatuko diona zenbat partida jokatu diren gehienez (*max*, *integer* motakoa). Datu hori eskatzean, egiaztatu behar da *max* > 3 dela betetzen dela, eta eskatzeko prozesua errepikatu behar da balio egokia lortu arte. Azpiprogramak balio hori itzuliko du.

- b) (0.75 puntu) **eskatu_jokaldia(...)** azpiprograma, erabiltzaileari jokaldi bat sar dezala eskatuko diona (*String* motakoa). Datu hori eskatzean, egiaztatu behar da sartutako balioa "harri ", "orri " edo "artazi" (6 karaktere letra xehez) dela. Eskatzeko prozesua errepikatuko da, jokaldirako baliozko balio bat lortu arte, eta hori izango da itzuliko dena.

jokaldi_random() funtzioa ez da programatu behar, baina erabili egin beharko da. Honakoa da bere espezifikazioa:

```
function jokaldi_random () return String is
-- Sarrera: -
-- Aurre: -
-- Irteera: String bat ordenagailuak jokatu duen aukera
           adierazten duena
-- Post: itzultitako balioa ausaz hautatzen du
--       "harri ", "orri ", eta "artazi" aukeren artean
begin
    ... - Implementazioa ez da inporta
end jokaldi_random;
```

- c) (1.75 puntu) **partida_jokatu(...)** azpiprogramak bi parametro jasotzen ditu (jokaldi1 eta jokaldi2, biak *String* motakoak), eta bi jokaldien arteko partidaren emaitza adierazten duen balio oso bat itzultzen du: 0 berdinketa izanez gero, 1 jokaldi1-ek irabaziz gero; edo 2 jokaldi2-k irabaziz gero.
- d) (1.75 puntu) **Programa nagusiak** jokatu diren gehieneko partida-kopurua (*max*) eskatuko diona erabiltzaileari, eta, ondoren, honela jokatu duena: **erabiltzaileak partida bat irabazten duenean, jokoa amaitu egiten da** eta erabiltzailea da irabazlea. Hala ere, behin *max* partida jokatuta, erabiltzaileak ez badu lortu bat ere irabaztea, orduan jokoa amaitu egingo da, eta kasu honetan irabazlea ordenagailua izango da. Kontuan izan behar da berdinketak jokatuak partidatzat hartzen direla. Programak adierazi beharko du nork irabazi duen, eta baita zenbat partida jokatu ondoren ere.

Froga-kasu baten adibidea:

Erabiltzailea	Ordenagailua
"orri "	"orri "
"artazi"	"orri "

PANTAILA
Erabiltzaileak jokoa irabazi du 2 partida jokatu ondoren.

Ariketa 3 – Haran-zenbakia (PYTHON): 3 puntu

Honako (azpi)programa hauen **espezifikazioa eta Python-eko implementazioa** eskatzen dira:

- (1 puntu) **haran_zenbaki()** azpiprograma: zenbaki oso bat emanez gero, haran-balioa den ala ez adieraziko du; hau da, 100 baino handiagoa dela, digitu-kopuru bakoitia duela, eta zenbaki horren erdiko digitua zenbakia osatzen duten gainontzeko digituak baino txikiagoa dela. Adibidez: 105 eta 67457 haran-zenbakiak dira, haatik 9 eta 356 ez dira.
- (2 puntu) Aurreko azpiprogramako **proben programa**. Implementazioa laburtzeko, proba-kasu orokor bakarra ezar dezakezue, betiere probatu behar diren gainerako kasu guztiak adierazten badira (adibidez, lehen laborategietan proba-kasuak zehazteko erabili zirenen bezalako taula baten bidez).