# INFORMATIKAREN OINARRIAK - AZTERKETA

**Irailak 15, 2010**

## *1. ARIKETA: KOSINUAREN KALKULUA (3 puntu)*

Radianetan emandako *x* angelu bat eta [0.000001, 0.1] tartean dagoen *r* zenbaki erreala eskatu ondoren, *x*-ren kosinua hurreratzen duen programa idatzi nahi da Taylor-ren formula erabiliz eta *r*-k zehaztuko doitasuna kontutan harturik. Taylor-ren formula hurrengoa da:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \frac{x^8}{8!} - \ldots \frac{x^{2n}}{2n!} = \sum_{i=0}^{n} (-1)^i \frac{x^{2i}}{(2i)!}$$

Programak *x*-ren kosinua hurbiltzen joango da hurbilketa baten balio absolutua eta aurreko hurbilketaren balio absolutuaren arteko diferentziaren balio absolutua *r* baino txikiagoa den arte. Adibidez, $c_i$ kosinuaren hurbilketa bat bada, eta $c_{i-1}$ aurreko hurbilketa bada, kalkuluaren prozesua amaituko da $| |c_i| - |c_{i-1}| |$ balio absolutua *r* baino txikiagoa denean.

**Adibidea:**
Suposatu dezagun erabiltzaileak angelu bezala 0.523598 (≈ π/6) eta doitasun bezala 0.08 sartu dituela, hau da, *x* = 0.523598 y *r* = 0.08. Programak kosinuaren hurbilketak kalkulatzen joango da azkenengo bi hurbilketen balio absolutuen arteko diferentziaren balio absolutua *r* baino txikiagoa den arte.

1. hurbilketa: $\cos(0.523598) = 1$

2. hurbilketa: $\cos(0.523598) = 1 - \frac{(0.523598)^2}{2!} = 0.862923$

3. hurbilketa: $\cos(0.523598) = 1 - \frac{(0.523598)^2}{2!} + \frac{(0.523598)^4}{4!} = 0.866054$

Hurbilketak kalkulatzearen prozesuak amaitzen du $| |0.866054| - |0.862923| |$ 0.08 baino txikiagoa delako.

Programa hau burutzeko hurrengo funtzioak inplementatu beharko dira:

a) **angelua_eskatu**: funtzio honek angelu bat radianetan adierazten duen zenbaki erreal bat eskatu eta programa nagusira itzultzen du. Funtzioak zenbakia [0, 6.283186] tartearen barruan dagoela egiaztatu behar du, kontrako kasuan berriro eskatuz zenbakia egokia izan arte. ***(0.25 puntu)***
b) **doitasuna_eskatu**: funtzio honek kosinua kalkulatzeko erabilik oden doitasuna adierazten duen zenbaki erreala eskatu eta programa nagusira itzultzen du. Funtzioak zenbakia [0.000001, 0.1] tartearen barruan dagoela egiaztatu behar du, kontrako kasuan berriro eskatuz zenbakia egoki aizan arte. ***(0.25 puntu)***
c) **faktoriala_kalkulatu:** funtzio honek, z zenbaki oso bat emanda, z-ren faktoriala kalkulatu eta itzultzen du ***(0.5 puntu)***
d) **terminoa_kalkulatu**: funtzio honek, *x* angelu bat radianetan eta *i* kalkulatu behar den termino zenbakia emanda, eta aurretik definitutako *faktoriala_kalkulatu* funtzioa erabiliz, i-ri dagokion Taylor-ren segidaren terminoa kalkulatu eta programa nagusira itzultzen du. ***(0.75 puntu)***
e) **hurbilketa_kalkulatu:** funtzio honek, x angelu bat radianetan eta r doitasuna emanda, Taylorren segida jarraituz eta aurretik definitutako *terminoa_kalkulatu* funtzioa erabiliz, x angeluaren kosinua r doitasunarekin kalkulatu eta funtzio nagusira itzultzen du. Funtzio honek terminoak kalkulatzen joango da, beraien arteko diferentzia r doitasuna baino txikiagoa izan arte. ***(0.75 puntu)***
f) **Programa nagusia**: programa nagusiak, aurretik definitutako funtzioak erabiliz, erabiltzaileari angelu bat eskatzen dio bere kosinua kalkulatzeko. Kalkulua burutzeko ezarri nahi den doitasuna ere eskatzen dio, dagokion hurbilketa kalkulatu eta pantailan bistaratzen du. ***(0.5 puntu)***

**Exekuzio adibidea:**

| | |
|---|---|
| Angelu bat radianetan sartu: *0.523598*<br>Nahi duzun doitasuna sartu ([0.000001, 0.1] tarteko zenbaki erreal bat): *2*<br>Adierazitako doitasuna ez da egokia.<br>Nahi duzun doitasuna sartu ([0.000001, 0.1] tarteko zenbaki erreal bat): *0*<br>Adierazitako doitasuna ez da egokia.<br>Nahi duzun doitasuna sartu ([0.000001, 0.1] tarteko zenbaki erreal bat): *0.08*<br><br>cos (0.523598) = 0.866054, 0.08-ko doitasunarekin<br>Jarraitzeko, tekla bat sakatu … | **Oharra**: teklatu bitartez sartutako datuak *letra etzan eta azpimarratuan* agertzen dira.<br>**OHARRA**: Behar diren aurredefinitutako funtzio matematiko guztiak erabili daitezke, trigonometrikoak izan ezik (cos, sin, …). |

## *2. ARIKETA: LABIRINTOAREN BIDE-ERAKUSLEA (3 puntu)*

Laberinto batean zehar mugitzen lagunduko digun programa bat idatzi nahi da. Erabiltzaileak zenbakien bidez zein aldetara joan behar den adierazi beharko du (aurrera, atzera, eskuinera, ezkerrera). Zenbakien eta norantzen arteko erlazioa honako hau da:

| | |
|---|---|
| 8: aurrera | 2: atzera |
| 4: ezkerrera | 6: eskuinera |

Erabiltzaileak emandako zenbakiak *derrigorrez* ***norabideak*** izena duen taula numeriko batean gorde beharko dira eta gero programak norantza bakoitzean zenbat urrats edo pausu eman behar diren esanez mezuak (testu moduan) aurkeztu beharko ditu:

Horretarako hurrengoa eskatzen da:

a) **hebideak_eskatu:** erabiltzileari norabideak zenbaki moduan sartzea eskatzen dio eta parámetro moduan lortutako array batean gordetzen ditu. Erabiltzaileak norabide egokiak sartu dituela egiaztatu behar da (8, 4, 6 edo 2). Norabide bate z den zenbaki bat sartzen bada, noarbide hori berriro eskatzen da. Norabide gehiago sartu nahi ez duela adierazteko, erabiltzaileak 0 sartuko du. Gehienez 100 norabide numeriko sartu ahal izango dira. ***(puntu 1)***

b) **helbideak_itzuli:** parámetro moduan hartzen duen norabideen numerikoen array batetik aurrera, argibideak testura itzuliko ditu adibideetan erakusten den moduan, eta pantailan bistaratuko ditu. Testurako itzulpenean berdinak diren zenbakien segidak kontutan izan beharko dira, norabide berean ematen diren segidako pausuak akumulatuz. Hau da, hiru zenbaki berdin segidan daudenean, norabide horretan hiru pausu eman behar direla adieraziko da. Itzulitako testuan ezin izango dira akumulatu segidan ez datozen zenbaki berdinak. ***(1.5 puntu)***

c) **funtzio nagusia:** *helbideak_eskatu* eta *helbideak_itzuli* funtzioak erabiliz, erabiltzaileari norabideak modu numerikoan sartzea eskatzen dio eta pantailan itzulitako argibideak bistaratuko ditu. ***(0.5 puntu)***

**Exekuzio adibidea:** (Teklatu bitartez sartutako datuak *letraetzan eta azpimarrautan* daude.)

Norabide bat sartu (2, 4, 6, 8): *5*
Norabidea ez da egokia. Norabide bat sartu (2, 4, 6, 8): *8*
Norabide bat sartu (2, 4, 6, 8): *4*
Norabide bat sartu (2, 4, 6, 8): *2*
Norabide bat sartu (2, 4, 6, 8): *4*
Norabide bat sartu (2, 4, 6, 8): *4*
Norabide bat sartu (2, 4, 6, 8): *6*
Norabide bat sartu (2, 4, 6, 8): *2*
Norabide bat sartu (2, 4, 6, 8)): *8*
Norabide bat sartu (2, 4, 6, 8): *8*
Norabide bat sartu (2, 4, 6, 8): *0*
Labirintoaren gida: 1 aurrera, 1 ezkerrera, 1 atzera, 2ezkerrera, 1eskuinera, 1 atzera , 2 aurrera.

***3. ARIKETA: KARAKTERE ORDEZKAKETA (4 puntu)***

Euskerazko testu batean letra bakoitza zenbat aldiz agertzen den aztertu nahi da, gero esperimentu bat egin eta karaktere ordezkaketa irakurgarritasunean duen eragina ikusteko.

Horretarako, **alfabetoa** izeneko string-a erabiliko dugu – OHARRAK begiratu-. String honek posizio bakoitzean alfabetoaren letra minuskula bat gordeko du alfabetikoki ordenatuta (lehenengo posizioan 'a', bigarrengoan 'b' eta abar). **agerpenak** izeneko taula numerikoa ere erabiliko dugo. Taula hau lehenengo testuan letra bakoitza zenbat aldiz agertzen den kontatzeko erabiliko da. Taula honetako posizio bakoitzean **alfabetoa** izeneko taulan posizio berean dagoen letraren agerpen-kopurua gordeko da. Hau da, *agerpenak* taularen i posizioan, *alfabetoa* taularen i posizioan agertzen den letra zenbat bider agertzen den testuan gordetzen da.

Jarraian aipatzen diren funtzioak definitu behar dira:

a) **karakterea_kontatu**: Parametro edo datu bezala karaktere-kate bat eta karaktere bat emanda, karaktere hori karaktere-katean zenbat aldiz agertzen den kalkulatu eta itzultzen duen funtzioa. ***(0.5 puntu)***

b) **karakterea_ordezkatu**: Parametro edo datu bezala **alfabetoa** izeneko string-a eta dagoeneko beteta egongo den **agerpenak** izeneko array-a emanda, erabiltzaileari karaktere-kate bat eskatu eta **agerpenak** taulan agerpen-kopuru handieneko karaktere bezala agertzen den karakterea agerpen-kopuru txikienekoaz ordezkatuz beste kate bat osatu eta pantailan aurkezten duen funtzioa. ***(2 puntu)***

c) **main**: Funtzio honek, erabiltzaileari *ereduzko testua* idaztea eskatuko dio (gehienez 100 karaktere) eta testu honetatik aurrera eta *karakterea_kontatu* funtzioa erabiliz, alfabetoko karaktere bakoitzaren agerpenak kontatuz eta emaitzak *agerpenak* taulan gordez joan beharko da. Ondoren, *karakterea_ordezkatu* funtzioa erabiliz, erabiltzaileari beste testu bat eskatuko zaio (gehienez 100 karaktere) eta kate berri bat kalkulatzen da *agerpenak* taulan dauden datuak kontuan hartuz maiztasun handieneko karakterea maiztasun txikienekoaz ordezkatuz. Amaitzeko, kate berria aurkeztuko da pantailan. ***(1.5 puntu)***

OHARRAK:

- Ariketan aipatzen den alfabetoa katea izateko, ondorengo katea hasieratuko da programan:

```
char alfabetoa[]="abcdefghijklmnñopqrstuvwxyz";
```

***alfabetoa***

| 0 | 1 | 2 | 3 | 4 | 5 | … | 24 | 25 | 26 | 27 |
|---|---|---|---|---|---|---|---|---|---|---|
| **'a'** | **'b'** | **'c'** | **'d'** | **'e'** | **'f'** | **…** | **'x'** | **'y'** | **'z'** | **'\0'** |

  - Alfabetoko 27 minuskulekin bakarrik arituko gara, zuriunea eta beste karaktereak ez dira kontuan hartu behar.
- Kontuan izan **agerpenak** izeneko taulan 0 balioa duten posizioek posizio horiei dagozkien letrak hasierako testuan ez direla agertu adierazten dutela eta horregatik maiztasun handiena eta txikiena kalkulatzerakoan ez dira hartu behar kontuan.
- Agerpen-kopuru handiena edota txikiena karaktere bati baino gehiagori badagokie, beraietako edozein aukera daiteke ordezkaketak egiterakoan.

**Exekuzio adibidea:** (Teklatu bitartez sartutako datuak *letra etzan eta azpimarrautan* daude.)

ereduzko katea sartu: *ez da bada*
Bigarren testua sartu: *gau izartsua*
Ordezkaketa eginda lortutako katea: **gbu izbrtsub**

Programa garatzeko erabili behar diren bost taulak (bat numerikoa eta 4 karakterezkoa), adibide honetan hórrela geratuko lirateke:

***alfabetoa***

| 0 | 1 | 2 | 3 | 4 | 5 | … | 24 | 25 | 26 | 27 |
|---|---|---|---|---|---|---|---|---|---|---|
| **'a'** | **'b'** | **'c'** | **'d'** | **'e'** | **'f'** | **…** | **'x'** | **'y'** | **'z'** | **'\0'** |

***eredua***

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | … | 100 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **'e'** | **'z'** | **' '** | **'d'** | **'a'** | **' '** | **'b'** | **'a'** | **'d'** | **'a'** | **'\0'** | **…** | |

***agerpenak***

| 0 | 1 | 2 | 3 | 4 | 5 | … | 24 | 25 | 26 |
|---|---|---|---|---|---|---|---|---|---|
| **3** | **1** | **0** | **2** | **1** | **0** | **…** | **0** | **0** | **1** |

Gehien agertzen den karakterea ‘a’ da (3 bider) eta gutxien agertzen dena ‘b’ edo ‘z’ (behin).

Orain suposatuko dugu erabiltzaileak “gau izartsua” testua tekleatzen duela, ordezkaketa egiteko:

***testua***

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | … |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 'g' | **'a'** | 'u' | **' '** | 'i' | 'z' | **'a'** | 'r' | 't' | 's' | 'u' | **'a'** | '\0' | … |

Adibide honetan ‘b’ letra aukeratu da, beraz, ‘a’ letraren agerpenak ‘b’-z ordezkatuko dira eta ‘b’ letrarenak, ‘a’-z. Kate berria *“gbu izbrtsub”*

***ordezkatua***

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | … |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 'g' | **'b'** | 'u' | **' '** | 'i' | 'z' | **'b'** | 'r' | 't' | 's' | 'u' | **'b'** | '\0' | … |