

Oinarrizko Programazioa.

4. laborategia. Sekuentziak (Adaz eta Pythonez)

Izena: _____ Data: _____

Laborategi honetan sekuentziak landuko dira.

1. ariketa

10 zenbaki osokoz osatutako sekuentzia bat edukita, sekuentzian dagoen lehenengo zenbaki bikoitia eta berau aurkitzen den posizioa inprimatzeko programa idatz ezazue.

1. Espezifikazioa

Sarrera: 10 zenbaki osoko sekuentzia

Aurre: sekuentzia 10 zenbakiz osatuta dago, ez bat gutxiago, ez bat gehiago.

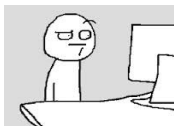
Irteera: zenbaki osoko bi

Post: balioa1 zerrendako lehenengo zenbaki bikoitia izango da eta balioa2 bera aurkitzen den posizioa. Balioa2 zenbakiak 0 balioa izango du baldin eta zerrendan ez badago zenbaki bikoitirik.

2. Proba kasuak

3. Algoritmoa

4. Simulazioa

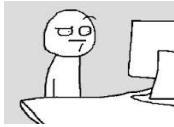


unibertsitatea



Universidad
del País Vasco

Euskal Herriko
Unibertsitatea



2. ariketa

10 zenbaki osokoz osatutako sekuentzia bat edukita, sekuentziako elementu guztiak goranzko ordena jarraitzen duten ala ez esango digun algoritmoa idatz ezazue.

1. Espezifikazioa

Sarrera: 10 zenbaki osoko sekuentzia

Aurre: sekuentzia 10 zenbakiz osatuta dago, ez bat gutxiago, ez bat gehiago.

Irteera: Mezu bat

Post: Pantailan “sekuentzia gorakorra” idatziko da baldin eta elementu **guztiak** goranzko ordena jarraitzen badute, bestela “sekuentzia gorakorra EZ” idatziko da

2. Proba kasuak

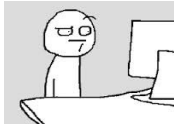
1, 2, 5, 7, 8, 9, 10, 12, 25, 67 → Bai, goranzko ordena jarraitzen du

1, 2, 5, 7, 8, 9, 10, 12, 25, 3 → Ez, ez du goranzko ordena jarraitzen

.....

3. Algoritmoa

4. Simulazioa



3. ariketa

10 zenbaki osokoz osatutako sekuentzia bat edukita, sekuentzian dauden elementuen artean, sekuentziako azken elementuagatik zatigarriak direnak inprimatzeko algoritmoa idatz ezazu.

1. Espezifikazioa

Sarrera: 10 zenbaki osoko sekuentzia

Aurre: sekuentzia 10 zenbakiz osatuta dago, ez bat gutxiago, ez bat gehiago.

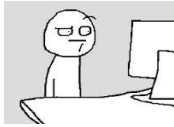
Irteera: Hainbat zenbaki osoko ala ezer ez.

Post: irteerako zenbaki bakoitza / sekuentziako azken balioa, hondarra = 0 da.

2. Proba kasuak

3. Algoritmoa

4. Simulazioa



4. ariketa

Eskatu erabiltzaileari zenbaki osoko bat (balio >0) eta irudika ezazu hurrengo grafikoa algoritmo bidez:

N=5 balitz

```
*0000
**000
***00
****0
*****
```

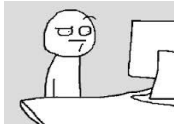
5. ariketa

OHARRA: Hemendik aurrera gure algoritmoak ADAz eta/edo Pythonez idatziko ditugu.

Orain gure lehenengo programa egikarituko dugu programazio-lengoaia batean. Bertan beharko ditugun konpiladore eta interprete guztiak egongo dira (alegia, ADA konpiladorea eta Python interpretea). Programatzeko erabiliko dugun testuinguru grafikoa ere bertan egongo da (eclipse IDE-a). Ala ere Pythonez programatzeko CodeSkultor erabiltzea errazena da, ez baita instalatu behar. Erabiltzeko joan hurrengo estekara (konprobatu Cisco VPN instalatuta duzula):

<http://www.codeskulptor.org/>

kaixoMundua programa egikarituko dugu bai ADA-z eta Python-ez. (“Kaixo mundua” mezua pantailan inprimatzen duen programa)



6. ariketa

Erabiltzaileari >0 den zenbaki oso bat eskatuko dion programa bat idatziko dugu. Programa horren helburua, erabiltzaileak sartutako zenbakiak zenbat digitu bakoiti dituen kalkulatzea izango da.

1. Espezifikazioa

Sarrera : zenbaki bat

Aurre: zenbaki oso bat >0

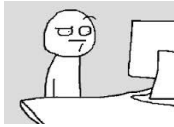
Irteera: zenbaki oso bat

Post: pantailatik ≥ 1 den balio bat inprimatuko da. Balio hori, sarrerako zenbakiaren digitu bakoitien kopurua izango da.

2. Proba kasuak

3. Algoritmoa

4. Simulazioa



7. ariketa

Erabiltzaileari 1tez hasten den zenbaki bitar bat eskatu (osoko gisa gordeko dena).
Kalkulatu zenbaki horren baliokide hamartarra.

1. Espezifikazioa

Sarrera : zenbaki bat

Aurre: 1z hasten den zenbaki bitar bat (hau da 0z eta 1z soilik osatua)

Irteera: zenbaki oso bat

Post: Sarrerako zenbaki bitarraren balio hamartarra. Gogoratu zenbaki bitarretan zifra bakoitzak bere pisua duela ($\text{zifra} * 2^{\text{posizioa}}$). Adibidea

$$10010 = 1 * 2^4 + 0 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = 18$$

ADAz berreketa: `oinarria**exp`

Python-ez berreketa: `oinarria**exp`

2. Proba kasuak

3. Algoritmoa

4. Simulazioa