

Oinarrizko Prog – 2. laborategiko Informea

Izena: Eneko, Gontzal eta Markel **Data:** 2014/10/01



Ikasitako kontzeptuak eta lortutako gaitasunak

Behin laborategia bukatu dudala, eskatu dizkidaten ariketetan erabiltzen diren kontzeptuak ondo ulertu ditudala kontsideratzen dut, eta beraz ariketak errepikatzeko gai izango nintzatekela esan dezaket. Erantzun hurrengo galdera-sorta esanez ea gai zaren (ala ez) hor galdetzen dena egiteko.

	Eneko	BAI	EZ
1	Espezifikazioa eta bere garrantzia ulertzen ditut	Bai	
2	Beharrezkoak diren proba kasuak proposatzeko gai naiz (behar diren guztiak, eta erredundantziarik gabe).	Bai	
3	Baldintzak erabilia proba kasu guztientzat emaitza egokia sortzeko gai naiz.	Bai	
4	Eraginkorra den soluzio bat proposatzeko gai naiz (hau da, beharrezkoak baino ez diren baldintzak ebaluatzen dituen algoritmoa).	Bai	
5	Algoritmo argi bat sortzerko gai naiz	Bai	
6	Algoritmo bati izen bat emateko gai naiz, gero behar den guztietan izen hori erabilia, algoritmo hori egikaritzeko gai naiz	Bai	
8	BESTELAKOAK: Azaldu zeintzuk		

Ezezeko erantzunen bat eman baduzu mesedez jarraian azaldu zergaitia, eta ze nolako neurria hartuko duzun sortu zaizun arazoa konpontzeko.

	Gontzal	BAI	EZ
1	Espezifikazioa eta bere garrantzia ulertzen ditut	Bai	
2	Beharrezkoak diren proba kasuak proposatzeko gai naiz (behar diren guztiak, eta erredundantziarik gabe).	Bai	
3	Baldintzak erabilia proba kasu guztientzat emaitza egokia sortzeko gai naiz.	Bai	
4	Eraginkorra den soluzio bat proposatzeko gai naiz (hau da, beharrezkoak baino ez diren baldintzak ebaluatzen dituen algoritmoa).	Bai	
5	Algoritmo argi bat sortzerko gai naiz	Bai	
6	Algoritmo bati izen bat emateko gai naiz, gero behar den guztietan izen hori erabilia, algoritmo hori egikaritzeko gai naiz	Bai	
8	BESTELAKOAK: Azaldu zeintzuk		

	Markel	BAI	EZ
1	Espezifikazioa eta bere garrantzia ulertzen ditut	Bai	
2	Beharrezkoak diren proba kasuak proposatzeko gai naiz (behar diren guztiak, eta erredundantziarik gabe).	Bai	
3	Baldintzak erabilita proba kasu guztientzat emaitza egokia sortzeko gai naiz.	Bai	
4	Eraginkorra den soluzio bat proposatzeko gai naiz (hau da, beharrezkoak baino ez diren baldintzak ebaluatzen dituen algoritmoa).	Bai	
5	Algoritmo argi bat sortzerko gai naiz	Bai	
6	Algoritmo bati izen bat emateko gai naiz, gero behar den guztietan izen hori erabilita, algoritmo hori egikaritzeko gai naiz	Bai	
8	BESTELAKOAK: Azaldu zeintzuk		



Atazen soluzioa

1.ARIKETA

Espezifikazioa

Enuntziatua anbigua al da? Bai

Bete

Sarrera: 3 zenbaki oso

Aurre-baldintza: $00 \leq \text{ordua} \leq 23$

$00 \leq \text{min}, \text{seg} \leq 59$

Irteera: 4 zenbaki oso

Post-baldintza: $00 \leq \text{ordua} \leq 23$

$00 \leq \text{min}, \text{seg} \leq 59$

Ordua, min, seg sarrerako balioen hurrengo segundua adieraziko dute. Egunez pasatuz gero eguna aldagaiak 1 balioko du, bestela 0.

Proba Kasuak

Sarrera

Irteera

Ordua 10 min 30 seg 10	Hurrengo segundua 10:30:11	Eguna: 0
Ordua 10 min 59 seg 10	Hurrengo segundua 10:59:11	Eguna: 0
Ordua 10 min 30 seg 59	Hurrengo segundua 10:31:00	Eguna: 0
Ordua 23 min 59 seg 59	Hurrengo segundua: 00:00:00	Eguna: 1
Ordua 22 min 59 seg 01	Hurrengo segundua 22:59:02	Eguna:0

Algoritmoa

orduak, minutuak, segunduak, eguna:integer;

eguna = 0;

Idatzi("Idatzi segunduak: ");

Irakurri(segunduak);

Idatzi("Idatzi minutuak: ");

Irakurri(minutuak);

Idatzi("Idatzi orduak: ");

Irakurri(orduak);

segunduak = segunduak + 1;

Baldin (segunduak > 59)

 segunduak = 0;

 minutuak = minutuak + 1;

 Baldin (minutuak > 59)

 minutuak = 0;

 orduak = orduak + 1;

 Baldin (orduak > 23)

 orduak = 0;

 eguna = eguna + 1;

 Amaitu

 Amaitu

Amaitu

Idatzi("&segunduak:&minutuak:&orduak, eguna &eguna);

Simulazioa

seg	min	ord	eguna59
59	59	23	00
00	00	00	01
-	-	-	-
01	59	22	00
02	59	22	00
-	-	-	-
59	30	10	00
00	31	10	00

2.ARIKETA:

Espezifikazioa

Enuntziatua anbigua al da? Bai

Bete

Sarrera: 6 zenbaki oso

Aurre-baldintza: $00 \leq \text{ordua} \leq 23$

$00 \leq \text{min}, \text{seg} \leq 59$

Irteera: 4 zenbaki oso

Post-baldintza: $00 \leq \text{ordua} \leq 23$

$00 \leq \text{min}, \text{seg} \leq 59$

Ordua, min, seg sarrerako balioen hurrengo segundua adieraziko dute. Egunez pasatuz gero eguna aldagaiak 1 balioko du, bestela 0.

Proba Kasuak

Sarrera1	Sarrera2	Gehiketa
ordua 10 min 30 seg 30	ordua 10 min 30 seg 30	21:01:00 eguna 0
ordua 10 min 40 seg 30	ordua 10 min 30 seg 30	21:11:00 eguna 0
ordua 20 min 50 seg 10	ordua 11 min 10 seg 2	08:00:12 eguna 1
ordua 10 min 50 seg 30	ordua 13 min 9 seg 30	00:00:00 eguna 1
ordua 23 min 50 seg 30	ordua 23 min 49 seg 30	23:40:00 eguna 1

1) Algoritmoa

```
orduak1, minutuak1, segunduak1, eguna:integer;  
orduak2, minutuak2, segunduak2, :integer;  
ordu_osoak, minutu_osoak, segundu_osoak, :integer;
```

```
eguna = 0;
```

```
Idatzi("Idatzi lehen orduaren segunduak: ");  
Irakurri(segunduak1);
```

```
Idatzi("Idatzi lehen orduaren minutuak: ");  
Irakurri(minutuak1);
```

```
Idatzi("Idatzi lehen orduaren orduak: ");  
Irakurri(orduak1);
```

```
Idatzi("Idatzi bigarren orduaren segunduak: ");  
Irakurri(segunduak2);
```

```
Idatzi("Idatzi bigarren orduaren minutuak: ");  
Irakurri(minutuak2);
```

```
Idatzi("Idatzi bigarren orduaren orduak: ");  
Irakurri(orduak2);
```

```
segunduak3 = segunduak1 + segunduak2;  
minutuak3 = minutuak1 + minutuak2;  
orduak3 = orduak1 + orduak2;
```


Baldin (segunduak3 > 59)

minutuak3 = minutuak3 + (segunduak3 / 60);

segunduak3 = (segunduak3 rem 60);

Amaitu

Baldin (minutuak3 > 59)

orduak3 = orduak3 + (minutuak3 / 60);

minutuak3 = (minutuak3 rem 60);

Amaitu

Baldin (orduak3 > 23)

eguna = eguna + (orduak3 / 24);

orduak3 = (orduak3 rem 24);

Amaitu

Idatzi("&segunduak3:&minutuak3:&orduak3, eguna &eguna3);

Simulazioa

seg1	min1	ord1	seg2	min2	ord2	seg3	min3	ord3	eguna
29	23	23	24	53	3	53	76	26	0
29	23	23	24	53	3	53	16	2	1
-	-	-	-	-	-	-	-	-	-
30	30	10	30	30	10	60	60	20	0
30	30	10	30	30	10	0	1	21	0
-	-	-	-	-	-	-	-	-	-
10	50	20	2	10	11	12	60	32	0
10	50	20	2	10	11	12	0	8	1

3.ARIKETA

Espezifikazioa

Enuntziatua anbigua al da? Bai

Bete

Sarrera: 3 zenbaki oso

Aurre-baldintza: $1 \leq \text{hilabetea} \leq 12$

$1900 \leq \text{urtea} \leq 2020$

Eguna hilabetearen menpean.

30 egun (4,6,9,11)

31 egun(1,3,5,7,8,10,12)

29 egun(2, bisestu denean)

28 egun (2, bisestu ez denean)

Urtea bisestu da 4gatik zatigarria denean eta 100gatik ez bada. Salbuespena: 400gatik zatigarria bada.

Irteera: 3 zenbaki oso

Post-baldintza: Urtea bisestu da 4gatik zatigarria denean eta 100gatik ez bada. Salbuespena: 400gatik zatigarria bada.

Proba Kasuak

Sarrera	Irteera
Eguna 20 hilabetea 1 urtea 2006	Hurrengo eguna 21 hilabetea 1 urtea 2006
Eguna 31 hilabetea 1 urtea 2006	Hurrengo eguna 1 hilabetea 2 urtea 2006
Eguna 30 hilabetea 1 urtea 2006	Hurrengo eguna 31 hilabetea 1 urtea 2006
Eguna 31 hilabetea 12 urtea	Hurrengo eguna 1 hilabetea 1 urtea 2007

2006	
Eguna 28 hilabetea 2 urtea 2006	Hurrengo eguna 1 hilabetea 3 urtea 2006
Eguna 28 hilabetea 2 urtea 2000	Hurrengo eguna 29 hilabetea 2 urtea 2000
Eguna 28 hilabetea 2 urtea 2013	Hurrengo eguna 1 hilabetea 3 urtea 2013
Eguna 13 hilabetea 12 urtea 2016	Hurrengo eguna 14 hilabetea 12 urtea 2016

Algoritmoa

eguna, eguna_max, hila, urtea: integer;

Irakurri(eguna);

Irakurri(hila);

Irakurri(urtea);

eguna = eguna + 1;

Baldin ((urtea rem 4) = 0 eta (urtea rem 100 != 0)) edo (urtea rem 400) = 0 orduan

 baldin hila = 2 orduan

 eguna_max = 29;

 Amaitu

Bestela

 Baldin hila = 2

 eguna_max = 28;

 Amaitu

Amaitu

Baldin hila = (1 edo 3 edo 5 edo 7 edo 8 edo 10 edo 12) orduan

eguna_max = 31;

bestela

eguna_max = 30;

Amaitu

Baldin (eguna > eguna_max) orduan

eguna = 1;

hila = hila + 1

Amaitu

Baldin (hila > 12) orduan

hila = 1;

urtea = urtea + 1;

Amaitu

Idatzi(“Hurrengo eguna &eguna hilabetea &hila urtea &urtea”);

Simulazioa

Eguna	Eguna_max	Hila	Urtea
31	kk	12	2003
1	31	1	2004
-	-	-	-
28	kk	2	2004
29	29	2	2004
-	-	-	-
28	kk	2	2005
1	28	3	2005

3. Laborategiko 2. Ariketa

Espezifikazioa

Bete

Sarrera: 2 zenbaki oso

Aurre-baldintza: $0 < n$, eta biderkatzailea ≤ 10

Irteera: Zenbaki oso bat

Post-baldintza: $0 < \text{emaitza:balioa} \leq 100 \mid \text{emaitza } n * \text{biderkatzailea}$
izango den

Proba Kasuak

Sarrera	Irteera
n 4 eta biderkatzailea 3	12
n 3 eta biderkatzailea 4	12
n 10 eta biderkatzailea 10	100
n 10 eta biderkatzailea 1	10

Algoritmoa

n, biderkatzailea, kontadorea, akumuladora:integer;

kontadorea $\leftarrow 0$;

akumuladorea $\leftarrow 0$;

Irakurri(n);

Irakurri(Biderkatzailea);

Errepikatu atera baldin eta (kontadorea = n)

akumuladorea \leftarrow akumuladorea + biderkatzailea;

kontadorea = kontadorea + 1;

amaitu;

Idatzi(akumuladorea);

Simulazioa

Buelta kop	Sartu?	n	Biderkatzailea	Kontadorea	Akumuladorea
0	-	7	4	0	0
1	Bai	7	4	1	4
2	Bai	7	4	2	8
3	Bai	7	4	3	12
4	Bai	7	4	4	16
5	Bai	7	4	5	20
6	Bai	7	4	6	24
7	Bai	7	4	7	28
8	Ez				