

Informatikaren Oinarriak-en AZTERKETA.

1 IIT MEKANIKA TALDEA 31. Ohiko deialdia.

2005-01-24

2004 - 05

IZEN ABIZENAK	NAN	NOTA

1
(0.25) Zenbat GigaByte dira $1024 * 2^{12}$ Kilobits?

2
(0.25) BUS-ari buruz dakizuna azaldu hitz gutxitan.

3
(1) 852,374 zenbakia zortzitarra izan daiteke?, eta hamartarra? Justificatu zure erantzuna.
Hamaseitarra dela suposatuz, bihurtu bi, zortzi eta hamar oinarrietara.

4
(0.5) Helbide-busak 24 hari eta memoriaren hitzak 32 bit baditu, zein izango da memoriaren tamaina MegaBytetan?

5
(1.5) Sortu C lengoian programa bat karaktere kate bat irakurrita palindromoa den edo ez den adierazten duena. (Adibideak: eme; kapak; dabalearrozalazorraelabad;)

6
(1.5)

```
x = 1;
while (x < 2)
{
    if (x % 10 == 1)
    {
        while (x < 3)
        {
            y = x ++;
        }
    }
    else
    {
        y = x - 1;
    }
    do
    {
        y = -- x;
    }
    while (x > 2);
}
printf("\n\tX = %d \n\tY = %d", x, y);
```

Programa hau analiza ezazu. Idatzi dagokion fluxu-diagrama eta zehaztu zer idazten duen pantailaz azken aginduk horrek.

8
(1) Funtzio bat sortu TAULA aldagaiaren LUZ luzera duen LER lerroan dauden zenbakien batura bueltatuko duena. LER eta LUZ sarrerako parametroak izango dira. TAULA bi dimentsiotako zenbaki osoen kanpo aldaia da. Adierazi nola erabili daitekeen funtzio hau ondorengo Bartzelona-Dakarrekoko luzera osoa kalkulatzeko, eta baita ere denbora totala kalkulatzeko.

7
(4)

Suposatu Bartzelona-Dakar lasterketan 5 kamioi, 5 kotxe eta 5 motor lehiatzen direla. 10 etapa dira eta datu hauek ezagutzen ditugu: kilometroak (harean eta errepidean), eta erreferentzi-denborak etapa bakoitzeko (orduak eta minutuak). Zehazki datu hauek ditugu.

Etapak	1	2	3	4	5	6	7	8	9	10
Km errep.	25	450	0	0	25	80	120	0	0	0
Km harean	0	0	350	500	470	135	480	780	630	50
Orduak	0	3	4	6	5	3	6	10	8	0
Minutuak	16	27	15	35	43	26	20	15	49	55

(Oharra: taula hau erazagutu programan eta ondoren erabili datuak sartuta egongo balira bezala)

Erreferentzi modura erabiltzen duten erregailua da: kamioiek 20 litro/100km, 15 litro/100km eta motorrek 10 litro/100km.

Sortu programa bat lasterketaren jarraipena egiten duena. Taula batean etapa eta lasterkari bakoitzeko datu hauek gorde: denbora (orduak eta minutuak), zenbat geldialdi (matxura dela, erregaiiaz hornitzeko dela, etab), erabilitako erregai litroak, eta puntuaketa.

Hasierako puntuaketa 0 izango dute, eta lasterketa bukatutakoan erreferentzi datuak gaindituz (puntu negatiboak) edo hobetuz (puntu positiboak) puntuak jasoko dituzte. Erreferentzi daturekiko: +/- 10 puntu lasterketa bukatzeko behar izan duen minutu bakoitzeko; +/-5 puntu litro bakoitzeko. Gainera -2 puntu geldialdi bakoitzeko.

Programak ondorengo egin beharko du:

1. Etapa bakoitzean lortu diren emaitzak eskatu eta gorde (denborak, geldialdi eta erregaia)
2. Lasterketa bukatutakoan lasterkari bakoitzeko puntuaketa totala kalkulatu.
3. Adierazi zein den lasterketaren etaparik luzeena.
4. Erakutsi pantailaz kamioien, kotxeen edo motorren puntuaketak, erabilitzailearen eskariz.

Informatikaren Oinarriak-en AZTERKETA.

1 IIT MEKANIKA TALDEA 31. Ohiko deialdia.

2005-01-21

2004 - 05

EBAZPENAK

1. Ariketa:

$1024 * 2^{12}$ Kilobit = $2^{10} * 2^{12}$ Kilobit = $2^{10} * 2^{12}$ Kilobit / 2^3 bit/Byte = 2^{19} KiloByte = 2^{19} KiloByte / 2^{20} kB/GgB = 2^{-1} GigaByte = 0.5 GigaByte

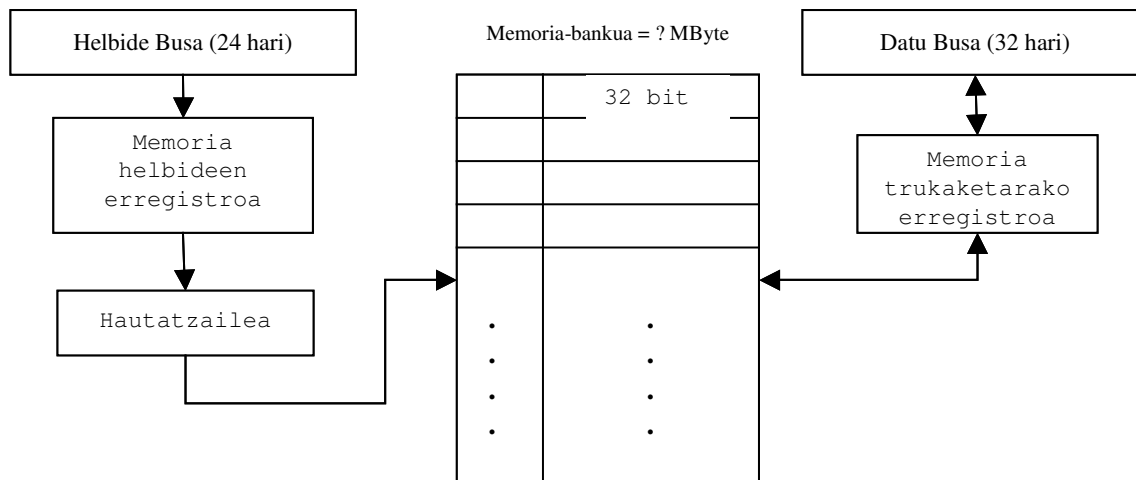
2. Ariketa:

3. Ariketa:

852,374 ezin da zortzitarra izan zeren erabili daitezkeen digituak 0, 1, 2, 3, 4, 5, 6 eta 7 dira. Aldiz, hamartarra izan daiteke 0 eta 9 arteko digituak erabiltzen direlako.

Hamaseitar	8	5	2	,	3	7	4		
Bitar	1000	0101	0010	,	0011	0111	0100		
Zortzitar	100	001	010	010	,	001	101	110	100
	4	1	2	2	,	1	5	6	4
Hamartar	$(852,374)_{16} = 8 * 16^2 + 5 * 16^1 + 2 * 16^0 + 3 * 16^{-1} + 7 * 16^{-2} + 4 * 16^{-3} = 8 * 256 + 5 * 16 + 2 * 1 + 3 * (1/16) + 7 * (1/256) + 4 * (1/4096) = 2048 + 80 + 2 + 0,1875 + 0,02734375 + 0,00097 = (2130,2158)_{10}$								

4. Ariketa:



24 hari helbide busan => 2^{24} posizio memoria-bankuan;

32 bit posizio bakoitzean => 4 Byte

Memoria-bankuaren tamaina: 2^{24} posizio * 4 Byte/posizio = $2^2 * 2^{24}$ Byte = 2^{26} bytes = 2^{26} Bytes * 1 Mbyte/ 2^{20} bytes = 2^6 Mbytes = 64 Mbytes.

5. Ariketa:

```
#include <stdio.h>
#include <conio.h>
#include <string.h>

void main()
{
    /* ALDAGAIEN ERAZAGUPENA */
    char esaldia [80];
    int hasierakoa, bukaerakoa, palindromoaDa;

    /* AGINDUAK */
    clrscr();

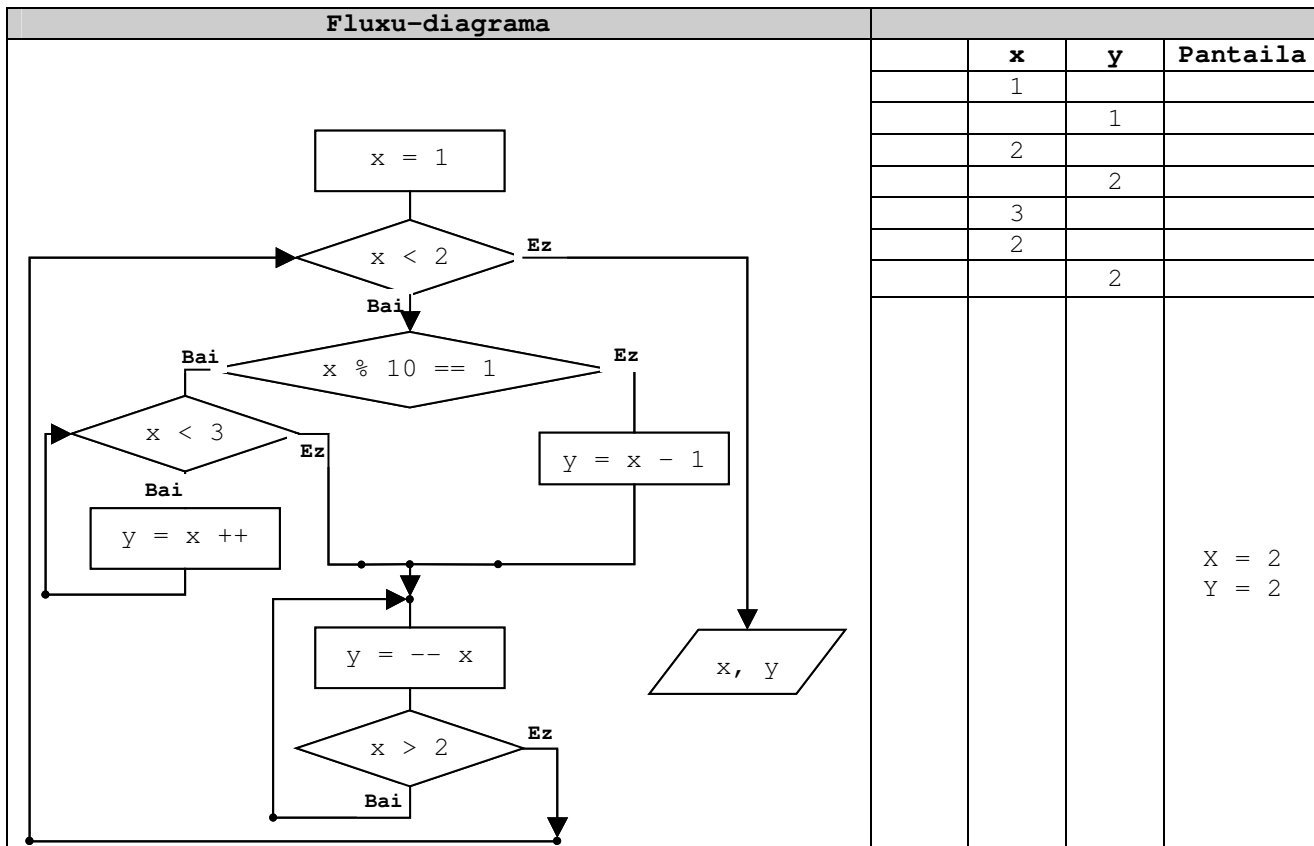
    /* IRAKURRI ESALDIA */
    printf("\n\n\tIdatzi esaldi bat: ");
    do
        {gets(esaldia);
        } while (strlen(esaldia) == 0);

    /* 2: KARAKTERE KATEA ANALIZATU */
    palindromoaDa = 1; /* PALINDROMOA DELA SUPOSATUKO DUGU */
    /* KARATERE KATEA NON BUKATZEN DEN JAKITEA */
    for (bukaerakoa=0; esaldia [bukaerakoa] != '\0'; bukaerakoa ++)
    {
        ;
    }
    bukaerakoa --; /* bukaerakoa: KATEAREN AZKEN POSIZIOA DUGU */

    /* HASIERAKO KARAKTEREAK BUKAERAKOEKIN ALDERATU */
    strlwr (esaldia);
    for (hasierakoa =0; hasierakoa < bukaerakoa && palindromoaDa; hasierakoa ++, bukaerakoa --)
    {
        if (esaldia[hasierakoa] != esaldia[bukaerakoa])
        {
            palindromoaDa = 0; /* EZ DA PALINDROMOA */
        }
    }

    if ( palindromoaDa)
    {
        printf("\n\n\tEsaldia palindromoa da.");
    }
    else
    {
        printf("\n\n\tEsaldia ez da palindromoa.");
    }
}
```

6. Ariketa:



7. Ariketa:

8. ariketan ikus daiteke bere erabilpena.

```
int BatuLerroa (int LER, int LUZ)
{int i, metatzailea;
  for (i=0, metatzailea = 0; i<LUZ; i++)
    {metatzailea += TAULA[LER][i];
    }
  return metatzailea;
}
```

Erabiltzeko era. Etapen datuak gordetzeko taula, TAULA izena badu:

Lasterketaren luzera:

Luzera = BatuLerroa (0, 10) + BatuLerroa (1,10);

Denbora minututan:

Denbora = BatuLerroa (2,10) * 60 + BatuLerroa (3, 10);

8. Ariketa:

```
#include <conio.h>
#include <stdio.h>
```

```
//PROTOTIPOAK
```

```
int BatuLerroa (int LER, int LUZ);
```

```
//KANPO ALDAGAIK
```

```
int TAULA[4][10]={ 25,450,0,0,25,80,120,0,0,0,
  0,0,350,500,470,135,480,780,630,50,
  0,3,4,6,5,3,6,10,8,0,
```

```

16,27,15,35,43,26,20,15,49,55}); //ETAPEN TAULA
//TAULA[datua][etapak]
//0: kilometroak errepedean
//1: kilometroak hanean
//2: erreferentzi denbora (orduak)
//3: erreferentzi denbora (minutuak)

void main()
{
int lasterkari, datua, balioa;
int metatzailea, auk, auk2, etapa;
int erreferentziMinutuak, minutuak, puntuak;
int max, posMax, topea, lerroa;
int lasterketa[5][15]; //LASTERKETAREN DATUAK KUDEATZEKO TAULA
//lasterketa[datua][lasterkari]
//0: denbora (orduak)
//1: denbora (minutuak)
//2: geldialdiak
//3: erregai litroak
//4: puntuaketa
//0-4: kamioiak
//5-9: kotxeak
//10-14: motorrak

clrscr();
do { //AUTOERREPIKAKORRA
//MENUA
printf ("\n\nMENUA:\n1. Etapen datuak gorde\n2. Puntuaketa totala\n");
printf ("3. Etaparik luzeena\n4. Puntuaketak aukeratu\n5. Irten\nHautatu");
scanf("%d", &auk);
while (auk<0 && auk > 5)
{puts ("Gaizki:\n");
printf ("MENUA:\n1. Etapen datuak gorde\n2. Puntuaketa totala\n");
printf ("3. Etaparik luzeena\n4. Puntuaketak aukeratu\n5. Irten\nHautatu");
scanf("%d", &auk);
}
switch (auk)
{case 1:
//1 LASTERKETA TAULAN GORDE DATUAK
{clrscr();
for (etapa=0; etapa<15; etapa++)
{for (lasterkari=0; lasterkari<15; lasterkari++)
{do {printf ("%d lasterkariak %d etapan zenbat ordu behar izan ditu? \n ",
lasterkari, etapa+1);
scanf ("%d",&balioa);
} while (balioa < 0);
lasterketa[0][lasterkari]=balioa;
do {printf ("%d lasterkariak %d etapan zenbat minutu behar izan ditu? \n
", lasterkari, etapa+1);
scanf ("%d",&balioa);
} while (balioa < 0 && balioa>59);
lasterketa[1][lasterkari]=balioa;
do {printf ("%d lasterkariak %d etapan zenbat geldialdi egin ditu? \n ",
lasterkari, etapa+1);
scanf ("%d",&balioa);
} while (balioa < 0);
lasterketa[2][lasterkari]=balioa;
do {printf ("%d lasterkariak %d etapan zenbat erregai erabili du? \n ",
lasterkari, etapa+1);
scanf ("%d",&balioa);
balioa = (balioa * 100) / (TAULA[0][etapa]+TAULA[0][etapa]);
} while (balioa < 0);
lasterketa[3][lasterkari]=balioa;

```

```

        }//for lasterkari bukaera
        }//for etapa bukaera
break;
} //case 1 bukaera

case 2:
//2: PUNTUAKETA TOTALA
{clrscr();
//ERREFERENTZI DENBORA KALKULATU. BATULERROA FUNTZIOA ERABILIKO DA.
erreferentziMinutuak = BatuLerroa(2,10) * 60 + BatuLerroa(3,10);

for (lasterkari = 0; lasterkari <15; lasterkari++)
    {minutuak = lasterketa[0][lasterkari] * 60;
    minuak += lasterketa[1][lasterkari];
    puntuak = (erreferentziMinutuak - minuak) * 10;
    puntuak += lasterketa[2][lasterkari] * (-2);

    if (lasterkari < 5) //KAMIOIAK
    {puntuak += ( 20 - lasterketa [3][lasterkari]) * 5;}
    else if (lasterkari < 10) //KOTXEAK
    {puntuak += (15 - lasterketa [3][lasterkari]) * 5;}
    else if (lasterkari < 15) //MOTORRAK
    {puntuak += (10 - lasterketa [3][lasterkari]) * 5;}
    lasterketa[4][lasterkari]= puntuak;
    }
break;
}

case 3:
//3: ETAPARIK LUZEENA.
{clrscr();
max = TAULA[0][0] + TAULA [1][0];
posMax = 0;
for (etapa=1; etapa<10; etapa ++))
    {if (TAULA[0][etapa] + TAULA [1][etapa] > max)
    {max = TAULA[0][etapa] + TAULA [1][etapa];
    posMax = etapa;
    }
    }
printf ("Etapa luzeena %d da, %d kilometroekin", posMax+1, max);
break;
}

case 4:
//4: KAMIOI, KOTXE, EDO MOTORREN PUNTUAK
{ //AZPIMENUA
clrscr();
printf ("AZPIMENUA:\n1. Kamioi\n2. Kotxe\n3. Motor\nNoren puntuak kontsultatu
nahi dituzu?");
scanf("%d", &auk2);
while (auk2 < 0 && auk2 > 3)
{puts ("Gaizki:\n");
printf ("AZPIMENUA:\n1. Kamioi\n2. Kotxe\n3. Motor\nNoren puntuak kontsultatu
nahi dituzu?");
scanf("%d", &auk2);
}
switch (auk2)
{case 1:{topea=5; break;}
case 2:{topea=10; break;}
case 3:{topea=15; break;}
}
for (lasterkari=topea-5; lasterkari < topea; lasterkari++)
{printf ("\n%d lasterkariak %d puntu ditu", lasterkari, lasterketa
[4][lasterkari]);

```

```

    }
    break;
}

} //switch bukaera

} while (auk!=5); //AUTOERREPIKAKORRA
puts ("\nSakatu tekla bat...");
getch();
} //main bukaera

////////////////////////////////////
/////FUNTZIOAK
////////////////////////////////////
int BatuLerroa (int LER, int LUZ)
{int i, metatzailea;
  for (i=0, metatzailea = 0; i<LUZ; i++)
    {metatzailea += TAULA[LER][i];
    }
  return metatzailea;
}

```