

7. LABORATEGIA

TAULA NUMERIKOAK / ARRAY-AK

1 HELBURUAK

Saio hau amaitzean, honetarako izando zara gai:

- Taula numerikoen funtzionamendua ulertu.
- Oinarrizko eragiketak burutu: taula bat deklaratu, taularen elemento baten irakurketa eta idazketa, taula bat zeharkatu, taula ordenatu,... funtzioak erabili gabe.

2 MOTIBAZIOA

- Demagun programa ba ebatzi nahi dugula, non, 0 batekin amaitzen den zenbaki segida bat emanda, alderantzizko ordenan aurkezten duena.
- Demagun programa ba ebatzi nahi dugula, non 0 batekin amaitzen den zenbaki segida bat emanda, gehien errepikatzen den zenbakia aurkezten duena.
- Aurreko arazoak eta beste batzuk gehiago ezin dira ezabatu taula numeriko hauek erabiltzeke.

2.1 Taula numerikoak c-z

- Taula numerikoak erabiltzeko deklaratu edo definitu behar dira, taularen identifikadorea, elementu kopurua eta datu tipoa adieraziz. Ad.: float nota_ikas[12];
- Taulako datu bat irakurri edo idazteko, irakurri edo idatziko den elementuaren indizea adierazi behar da. Ad.: taula baten lehenengo elementuarentzat:
 - Irakurketa -> printf("%d",nota_ikas[0]);
 - Idazketa -> scanf("%d",¬a_ikas[0]);
- Taula baten elementuak zeharkatzeko, 0-tik "elementu-kopurua - 1"-ra doan indizea erabilik dugu. Ad.: taula baten elementu guztien karga eta aurkezpena:

Ad:

```
/* programa nagusiaren hasiera */
void main()
{
    /*taula zeharkatzeko indizea */
    int i;

    /*taularen elementu kopurua */
    int const NELE=10;

    /*Zenbaki osoen taula baten definizio edo deklarazioa */
    int notak_ikas[NELE];

    /* taula baten elementu guztien karga */
    for (i=0;i<=NELE -1;i++)
```

```

    {
        printf("Sartu %d ikaslearen nota %d",i);
        scanf("%d",&notak_ikas[i]);
    }

    /* taulako elementu guztien aurkezpena */
    for (i=0;i<=NELE -1;i++)
    {
        printf("%d ikaslearen nota %d da",i,notak_ikas[i]);
    }
}

```

3 ARIKETAK

3.1 1. ariketa

3.1.1 Helburua:

1. ariketaren helburua taula bat bere elementuak tratatuz nola zeharkatzen den ikastea da, kasu honetan ezkererantz mugituz.

3.1.2 Laguntza:

Enuntziatua ebatzita ematen da gida bezala erabiltzeko 1. eta 2. enuntziatuentzat:

```

#include <stdio.h>
#include <stdlib.h>
void main()
{
    int t[10],ind,luz,zenb,aux;
    ind=0;

    do{
        printf ("Sartu zenbaki bat");
        scanf("%d",&zenb);
        t[ind]=zenb;
        ind=ind+1;
    }while(ind<10);

    /* bukletik irtetzean ind 10-ra apuntatzen dago, hau da, bektoreak daukan
    elementu kopurua. Lehenengo elementua aux-en gordetzen dugu, beste
    guztiak ezkererantz desplazatzen hasi baino lehen. */

    luz=ind;
    aux=t[0];

    printf ("aux %d\n",aux);

    /* Indizea 1-ra hasieratzen dugu, ezkererantz mugitzean, posizio bat hartu
    eta ezkerreko elementuari esleiten diogu, horregaitik hasten gara bigarren
    posizioarekin, lehenengoari esleiteko */

    ind=1;

```

```

/*Desplazamenturako buklea, KONTUZ, bi gauza kontutan hartu: ind=1-an
hasten gara eta ind=luz-1 ean amaitzen dugu*/

while(ind < luz){
    t[ind-1]=t[ind];
    ind=ind+1;
}

/* Azkenengo posizioan, hasieran lehenengoan zegoena gordetzen dugu */
t[luz-1]=aux;

/*Ez ahaztu indizea 0-ra jartzeaz taula zeharkatu eta inprimitzeko buklea
inplementatzeko */

ind=0;
while(ind < luz){
    printf("%d\n",t[ind]);
    ind=ind+1;
}
system("PAUSE");

```

3.1.3 1. enuntziatua:

Erabiltzaileari N tamainuko taula baten elementuak eskatzen dizkion eta ondoren elementu guztiak ezkererantz desplazatzen dituen programa bat egin, lehenengo elementua azkenengoan utziz.

Adibidea (demagun N 10 dela):

EMAN 10 ZENBAKI OSO: 3 24 7 -25 0 3 21 19 3 895

Array-a EZKERRERANTZ desplazatuta, honela geratzen da:
24 7 -25 0 3 21 19 3 895 3

3.2 2. ariketa

3.2.1 Helburua:

2. ariketaren helburua aurreko ariketan ikasitakoa praktikan jartzea da.

3.2.2 2. enuntziaturako laguntza:

1. ariketa erabili

3.2.3 2. enuntziatua:

Aurreko ariketan bezala, baina orain elementuak eskuinarantz desplazatuz eta azkenengo elementua lehenengo posizioan utziz.

Adibidea (demagun N 10 dela):

EMAN 10 ZENBAKI OSO: 3 24 7 -25 0 3 21 19 3 895

Array-a ESKUINERANTZ desplazatuta, honela geratzen da:

895 3 24 7 -25 0 3 21 19 3

3.3 3. ariketa

3.3.1 Helburua:

3. ariketaren helburua programa taula zeharkatzean igarotzen dituen egoerak gordetzen dituen indize eta kontadoreen erabilera da. Egoera moduan kasu honetan elementu berdinaren kopuru maximoa eta bere hasierako eta amaierako posizioak hartuko dugu.

3.3.2 3. enuntziaturako laguntza:

Taulako elementuak zeharkatzean, hurrengo elementua eta oraingoa ezberdink direnean, sekuentzia batekin amaitu dugu eta, kasua bada, errepikatutako posizioen maximoa eta indizeak eguneratuko ditugu.

3.3.3 3. enuntziatua:

N elementuko taula bat emanda, kalkulatu zein den jarraian doazen elementu errepikatuen kopurua, eta baita ere bere hasiera eta amaierako posizioak.

Adibidea (demagun N 10 dela):

i:	0	1	2	3	4	5	6	7	8	9
V:	15	23	23	5	7	4	4	4	23	23

Jarraian agertzen diren errepikatutako elementuen kopuru handiena 3 da eta 5 eta 7 posizioen artean duade.

3.3.4 4. ariketa

3.3.5 Helburua:

4. ariketaren helburua taulak kargatu edo zeharkatzeko orduan baldintzak erabiltzen ikastea da, bereziki, beti elementu kopuru berdina es daukagulako.

3.3.6 4. enuntziaturako laguntza:

Taula hasieratik azkenera zeharkatzen dugu, minimoa bilatuz eta bai bere balioa bai bere posizioa gordez.

3.3.7 4. ariketaren enuntziatua

Erabiltzaileari zenbaki segida bat eskatu eta N luzerako array batetan gorde. Programak zenbakiak eskatu behar izango ditu, taulan N elementu sartu arte edo reabiltzaileak zenbaki negatibo bat teklatu arte. Behin array-a gehienez N zenbaki positiboekin dagoela, txikiena lehenengo posizioan dagoenarekin trukatzeari eskatzen da.

Adibidea (demagun N 10 dela):

SARTU GEHIENEZ 10 ZENBAKI (NEGATIBOA AMAITZEKO):
45 67 13 29 3 28 32 51 -1

Array-a, behin trukea eginda, honela geratuko litzateke:
3 67 13 29 **45** 28 32 51

3.4 5. ariketa

3.4.1 Helburua:

5. ariketaren helburua taula baten elementuak handienatik txikienara ordenatzeko burbuila algoritmoa erabiltzea da.

3.4.2 5. enuntziatua:

Gehienez N elementu dituen bektore baten elementuak handienatik txikienara ordenatzen dituen programa bat egin. Aurreko ariketan bezala, bektorea bete barik gelditu daiteke, erabiltzaileak zenbaki negatibo bat teklatzen badu, balio gehiago eskatuko ez direlako.

Adibidea (demagun N 10 dela):

SARTU GEHIENEZ 10 ZENBAKI (NEGATIBOA AMAITZEKO):

4.2 1.3 4.75 4.2 0.55 -1.3

Array-a, behin trukea eginda, honela geratuko litzateke:

.55 1.3 3.25 4.2 4.2 4.75