
// Μέσος όρος τιμών πίνακα που είναι μικρότεροι ή ίσοι από έναν αριθμό (100)

```
#include <stdio.h>
const int max_grade=100;
int grades[10];
int idx;
int total=0;
main()
{
    for(idx=0;idx<10;idx++)
    {
        printf("enter grade for person %d:",idx+1);
        scanf("%d",&grades[idx]);

        while(grades[idx]>max_grade)
        {
            printf("\nThe highest grade is %d",max_grade);
            printf("\nenter correct grade:");
            scanf("%d",&grades[idx]);
        }
        total +=grades[idx];
    }
    printf("\n The average is %d\n",(total/10));
    return 0;
}
```

// Εμφάνιση διδιάστατου πίνακα

```
#include <stdio.h>
#include <stdlib.h>
void disp(int v[3][4]);

void main()
{
    int a[3][4]={{1,2,3,4},{5,6,7,8},{9,1,2,3}};
    disp(a);
}

void disp(int v[3][4])
{
    int i,j;
    for (i=0;i<3;i++)
    {
        for (j=0;j<4;j++)
        {
            printf("%d ",v[i][j]);
        }
        printf("\n");
    }
}
```

// Εύρεση ενός αριθμού σε έναν πίνακα

```
#include <stdio.h>
int ar,id,f,pos;
int pin[10]={5,10,3,4,20,1,11,19,7,9};
f=0;
void main ()
{
    printf("doste arithmo ");
    scanf("%d",&ar);
    for (id=0;id<10;++id)
    {
        if (pin[id]==ar) {
            f=1;
            pos=id;
            break;
        }
    }
    if (f==1)
        printf("uparxi o arithmos %d sti thesi %d",ar,pos);
    else
        printf("den yparxi o arithmos %d ",ar);
}
```

// βρισκει τη θέση της μικροτερης τιμης απο στοιχεία ενός πίνακα

```
#include <stdio.h>
const int max=10;
int array[max],count;
int func(int x[],int y);
void main()
{
    for (count=0;count<max;count++)
    {
        printf("enter an integer value ");
        scanf("%d", &array[count]);
    }
    printf("\nthe position of min is %d ", func(array,max));
}
int func(int a[],int y)
{
    int i,v,pos;
    pos=0;
    for (i=0,v=a[0];i<y;++i)
    if (v>a[i])
    {
        v=a[i];
        pos=i;
    }
    return pos;
}
```

// Ανάλυση μισθού σε νομίσματα

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

```
void main()
```

```
{
```

```
    int ax_nom[12]={ 10000,5000,1000,500,200,100,50,20,10,5,2,1};
```

```
    int pl_nom[12];
```

```
    int mist,pil,i;
```

```
    for (i=0;i<12;i++)
```

```
        pl_nom[i] = 0;
```

```
    printf("doste mistho ");
```

```
    scanf("%d", &mist);
```

```
    for (i=0;i<12;i++)
```

```
    {
```

```
        if (mist < ax_nom[i])
```

```
            continue;
```

```
        else
```

```
        {
```

```
            pil=mist/ax_nom[i];
```

```
            pl_nom[i]=pil;
```

```
            mist=mist-pil*ax_nom[i];
```

```
            if (mist == 0)
```

```
                break;
```

```
        }
```

```
    }
```

```
    for (i=0;i<12;i++)
```

```
    {
```

```
        if (pl_nom[i] != 0)
```

```
            printf("%d x %d \n",pl_nom[i],ax_nom[i]);
```

```
    }
```

```
}
```

// Εισαγωγή μήνα και έτους και εμφάνιση ημερών του μήνα

```

#include <stdio.h>
#include <string.h>
int year;
char
*months[12]={ "jan","feb","mar","apr","may","jun","jul","aug","sep","okt","nov","de
c"};
int monthdays[12]={31,28,31,30,31,30,31,31,30,31,30,31};

int disekto(int year)
{
    if ((year % 4 ==0) && (year % 100 != 0) || (year % 400 == 0))
        return 1;
    else
        return 0;
}

int daysofmonth(char *monthname, int year) //int daysofmonth(char monthname[], int
year)
{
    int i;
    for (i=0;i<12;i++)
    {
        if (!strcmp(monthname,months[i]))
        {
            if (i==1)
                return monthdays[i]+disekto(year);
            else
                return monthdays[i];
        }
    }
    return 0;
}
void main()
{
    int days;
    char mname[3];
    printf("give a month and year ");
    scanf("%s %d", mname,&year);
    days = daysofmonth(mname,year);
    if (days != 0)
        printf ("the days of %s is %d",mname,days);
    else
        printf ("wrong month");
}

```
