

## 1. Példa

Írjon C programot a következő feladatra: A gets() függvénnyel billentyűzetrol bekert karaktervektorban meg kell számlálni a 'k' karakterek előfordulásának számát! Az eredményt írassa ki!

```
#include <stdio.h>
void main(void)
{
    char s[100];
    int i, db;
    puts("k karakterek számlálása\n\n");
    printf("Adja meg a szöveget:");
    gets(s);
    for (db = i = 0; s[i] != '\0'; i++)
    {
        if (s[i] == 'k')
        {
            db++;
        }
    }
    printf("\n\nA megadott szövegben %d darab 'k' karakter volt.",db);
}
```

## 2. Példa

Osszetett logikai feltételnek eleget tevő elemek megszámlálása. Egy fogazógéphez az áttételi viszonyt cserefogaskerekkel lehet beállítani. Egy előzetes vizsgálat annak a megállapítását kívánja, hogy a két csoportban (más furatmerettel, de azonos modullal rendelkező) megadott fogaskerek között hány olyan pár kepezhető, melyek fogszámhányadosa egész szám. Ezeket a párokat írassuk ki és számláljuk meg, ezek között hány olyan pár van, amelynél a fogszámhányados 3 egész számú többszöröse. A csoportokban 18-18 db fogaskerek van. Az elsőben a legkisebbnek 11, a másodikban a legkisebbnek 55 foga van.

```
#include <stdio.h>
#define db 18
#define min1 11
#define min2 55
int csoport1[db], csoport2[db];
void main()
{
    int i,j,parokszama, fogszámhanyados;
    puts("Fogaskerekpárok\n");
    // csoportok feltöltése:
    for (i=0; i<db; i++)
    {
```

```
        csoport1[i]= min1 + i;
        csoport2[i]= min2 + i;
    }
    // csoportok kiiratas:
    puts("Az első csoportban lévő kerek fogszámai:");
    for (i=0; i<db; i++)
    {
        printf("%4d", csoport1[i]);
    }
    puts("\nA második csoportban lévő kerek fogszámai:");
    for (j=0; j<db; j++)
    {
        printf("%4d", csoport2[j]);
    }
    printf("\n\nFolytatáshoz nyomjon Entert!");
    getch();
    // vizsgálat és számlálás:
    puts("A vizsgálat eredménye:\n");
    parokszama=0;
    for (i=0; i<db; i++)
    {
        for (j=0; j<db; j++)
        {
            if (csoport2[j] % csoport1[i] == 0)
            {
                fogszamhanyados= csoport2[j] / csoport1[i];
                printf("%d a fogszamhanyados a 2. csoport %d és az 1. csoport %d fogszáma között.\n",
                    fogszamhanyados, csoport2[j],csoport1[i]);
                // 3 többszöröseinek számlálása:
                if (fogszamhanyados % 3 == 0)
                {
                    parokszama++;
                }
            }
        }
    }
    printf("\n\nHárom többszöröset adó attetelu parok száma= %d",
        parokszama);
}
```

### 3. példa

Kiválasztás összetett logikai feltétel alapján. Válasszuk ki azt a vektorelemet egy numerikus, legalább negyelemű vektorból, amelyiknek a közvetlenül megelőző és a közvetlenül rákövetkező számokkal adódó átlaga legközelebb esik a sorozat átlagértékéhez.

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <time.h>
float V[200];
void main()
{
    int i,n,index;
    float atlag, atlag3, elteres, elteresmin;
    srand(time(NULL));
    puts("Kivalasztas atlagok alapjan\n");
    do
    {
        printf("\nElemek szama=");
        scanf("%d",&n);
    }
    while (n<4);
    //feltoltes veletlenszamokkal:
    for (i=0; i<n; i++)
    {
        V[i]= (rand() % 100) /1.5;
    }
    //Az osszes elem atlaganak szamitasa:
    for (atalag=0.0, i=0; i<n; i++)
    {
        atlag+= V[i];
    }
    atlag = atlag/n; //vagy atlag /= n;
    printf("\nA sorozat atlaga= %10.3f \n", atlag);
    // a kivalasztas:
    elteresmin= 1e20; // tiz a huszadikon, jo nagy
    for (i=1; i<n-2; i++) // masodiktol az utolso elottiig fut!
    {
        atlag3= (V[i-1]+V[i]+V[i+1])/3;
        elteres= (atalag>atalag3) ? atlag-atalag3 : atlag3-atalag;
        if (elteres<elteresmin)
        {
            elteresmin= elteres;
            index= i;
        }
        printf("\nA harom egymasutani szam: %10.3f %10.3f %10.3f",
            V[i-1],V[i],V[i+1]);
        printf("\nAtlaguk= %8.3f, az elteres= %8.3f", atlag3, elteres);
    }
    printf("\n\nA %d indexu szam es szomszedai atlaga ter el legkevesbe,
    az elteres erteke= %10.3f", index, elteresmin);
}
//Futtassuk 12, vagy kevesebb elemmel!
```

From:

<https://edu.iit.uni-miskolc.hu/> - **Institute of Information Science - University of Miskolc**

Permanent link:

[https://edu.iit.uni-miskolc.hu/tanszek:oktatas:szamitastechnika:tovabbi\\_algoritmusok](https://edu.iit.uni-miskolc.hu/tanszek:oktatas:szamitastechnika:tovabbi_algoritmusok)

Last update: **2022/09/05 21:51**

