Feladat: adott egy 8 elemű float vektor, adja össze a vektor elemeit és írja ki a konzolra.

Első megoldás, egyszerű ciklus:

```
#include <stdio.h>
int main()
{
    float vektor[] = {1.2, 2.3, 3.4, 4.5, 5.5, 6.7, 4.5, 0.4};
    float szumma = 0;
    const int elemszam = 8;
    for (int k = 0 ; k < elemszam; k++)
    {
        szumma += vektor[ k ];
    }
    printf("Szumma= %f", szumma);
}
```

Második megoldás, lebegőpontos mutatóval:

```
#include <stdio.h>
int main()
{
    float vektor[] = {1.2, 2.3, 3.4, 4.5, 5.5, 6.7, 4.5, 0.4};
    float szumma = 0;
    const int elemszam = 8;
    float *v = vektor;
    for (int k = 0 ; k < elemszam; k++, v++)
    {
        szumma += *v;
    }
    printf("Szumma= %f", szumma);
}
```

Harmadik megoldás, a k változót próbáljuk elhagyni:

```
    #include <stdio.h>
    int main()
    {
        float vektor[ ] = {1.2, 2.3, 3.4, 4.5, 5.5, 6.7, 4.5, 0.4};
        float szumma = 0;
        int elemszam = 8;
        float *v = vektor;
        for ( ; v != &vektor[elemszam]; v++)
        {
            szumma += *v;
```

```
        }
        printf("Szumma= %f", szumma);
    }
```

Negyedik megoldás, do-while ciklussal, valamit a pointer növelését ne elkülönítve végezzük:

```
    #include <stdio.h>
    int main()
    {
        float vektor[ ] = {1.2, 2.3, 3.4, 4.5, 5.5, 6.7, 4.5, 0.4};
        float szumma = 0;
        int elemszam = 8;
        float *v = vektor;
        do
        {
            szumma += *v;
        }
        while(++v != &vektor[elemszam]);
        printf("Szumma= %f", szumma);
    }
```

Fontos megjegyzés: itt nem baj hogy a vektor index túl van címezve (&vektor[elemszam]), mert pontosan arra a memóriaterületre mutat, amit már nem szabad használnia, azaz ha a **v** pointer értékét először megnövelve, majd öszehasonlítva az utolsó utáni vektorelem memóriacimével megegyezik, pontosan akkor nem lesz a ciklus többször végrehajtva.

Ötödik megoldás, próbáljuk meg az elemszám változót is elhagyni:

```
#include <stdio.h>
int main()
{
    float vektor[ ] = {1.2, 2.3, 3.4, 4.5, 5.5, 6.7, 4.5, 0.4};
    float szumma = 0, *v = vektor;
    int elemszam = sizeof(vektor) / sizeof(float);
    do
    {
        szumma += *v;
    }
    while(++v != &vektor[elemszam]);
    printf("Szumma= %f", szumma);
}
```

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:vektorok_peldak

Last update: **2023/11/29 08:05**