

Wojciech Myszka

Laboratorium 9: Tablice (i funkcje)

2016-05-07 09:04:39 +0200

1. Wprowadzenie

Zasadniczym tematem zajęć jest oskonalenie umiejętności operowania na tablicach (oraz używaniu ich jako parametrów funkcji).

Dodatkowym motywem jest tworzenie (w miarę) **uniwersalnych** funkcji (kiedy tylko istota tego co robią jest taka sama).

2. Statystyki tablicy.

Stosunkowo prosty program, który pyta użytkownika o wymiary tablicy dwuwymiarowej (liczbę wierszy, na przykład n i liczbę kolumn, na przykład m , a następnie wykonuje następujące czynności:

- Organizuje miejsce na tablicę i tablice pomocnicze (czyli deklaruje wszystkie dane).
- Wypełnia tablicę danymi Aby uprościć sobie życie zakładamy, że program wyposażony będzie w **osobną funkcję** wypełniającą tablicę losowymi danymi. Może ona wyglądać jakoś tak:

```
#include <stdlib.h>
#include <time.h>
...
{
    ...
    srand( time(NULL) );
    for ( i = 0; i < m; i++)
        for ( j = 0; j < m; j++ )
            tablica[i][j] = rand() % M;
}
```

(oczywiście, trzeba uzupełnić wszystko co potrzebne!)

- Następnie program wylicza średnią (**osobna funkcja!**) i wariancję (**osobna funkcja!**) dla każdego wiersza i każdej kolumny oraz dla całej tablicy i wyprowadza na terminal wszystkie wyniki.

Dostępne jest najprostsze ćwiczenie pozwalające przekazać tablicę dwuwymiarową do funkcji.

```
#include <stdio.h>
#include <stdlib.h>
void drukuj(int n, int m, int z[n][m])
{
    int i, j;
    for ( i = 0; i < n; i++ )
        for ( j = 0; j < m; j++ )
            printf("%d\n", z[i][j]);
}

int main(int argc, char **argv)
{
    int i, j;
    int N = 3;
    int M = 5;
    int a[N][M];
    for ( i = 0; i < N; i++ )
        for ( j = 0; j < M; j++ )
            a[i][j] = rand() % 20;
    for ( i = 0; i < N; i++ )
    {
        for ( j = 0; j < M; j++ )
            printf("%10d, ", a[i][j]);
        printf("\n");
    }
    drukuj(N, M, a);
    return 0;
}
```

Zadanie może być wykonane na dwa sposoby: korzystając z deklaracji tablic (jak w powyższym ćwiczeniu) albo przydzielając na nie pamięć w sposób dynamiczny (korzystając z funkcji malloc¹). Tu jest ściągą.

¹ Skorzystanie z funkcji malloc wymaga dodania wiersza `#include <stdlib.h>`.

3. Wersja PDF tego dokumentu...

... pod adresem.

Wersja: 50 z **drobnymi modyfikacjami!** data ostatniej modyfikacji 2016-05-07 09:04:39 +0200