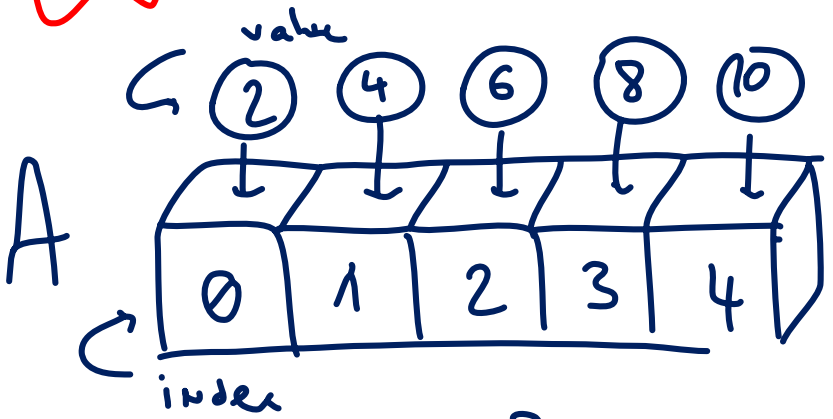


ЛНБС

# Массивы

AD



C++

```

dr. element
int A[5];
A[0] = 2;
A[1] = 4;
A[2] = 6;
↑
wert:

```

```

const int N = 5000;
int A[N];
for (int k = 0; k < N; k++)
  cin >> A[k];
for (int k = 0; k < N; k++)
  cout << A[k] * 2;

```

## Console App

- 1) # + R ↵
- 2) CMD ↵
- 3) CD Desktop
- 4) 2.exe > 2.txt
- 5) 1.exe < 2.txt
- 6) 1.exe < 2.txt > 1.txt

3) Задача:

Напишите 10000 чисел от результата на Фибоначчи

1 1 2 3 5 8 13 ...

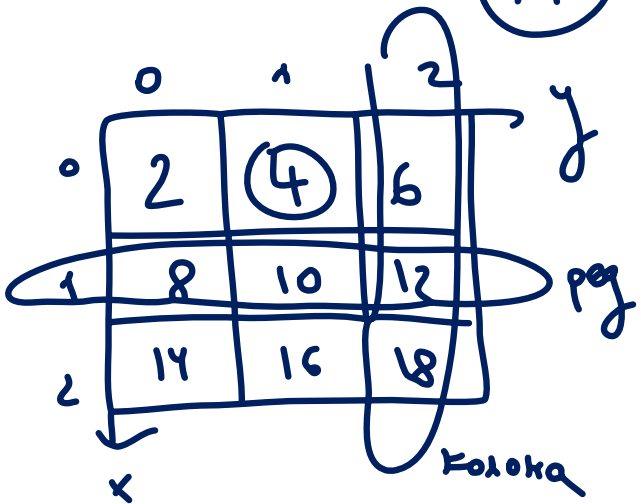
$$Fib[k] = Fib[k-2] + Fib[k-1]$$

↑ 1000



# Масиви

(A)



(2D) Array

Matrix

(C++)

```

int A[3][3];
A[0][1] = 4;
:

```

## Въвеждане.

```

for (int x=0; x<5; x++)
for (int y=0; y<3; y++)
cin >> A[x][y];

```

## Извеждане.

```

for (int x=0; x<3; x++) {
for (int y=0; y<3; y++)
cout << A[x][y] << " ";
}

```

## Задачи:

(4) Умножение на матрица с числ.

$$A \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \cdot 2 = A' \begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \\ 14 & 16 & 18 \end{bmatrix}$$

(5) Събиране на 2 матрици

$$A \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + B \begin{bmatrix} 9 & 8 & 7 \\ 6 & 5 & 4 \\ 3 & 2 & 1 \end{bmatrix} = C \begin{bmatrix} 10 & 10 & 10 \\ 10 & 10 & 10 \\ 10 & 10 & 10 \end{bmatrix}$$

(6) Умножение на матрици по "Адамор"

(7) Проверка на 2 матрици

$$A \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + B \begin{bmatrix} 9 & 8 & 7 \\ 6 & 5 & 4 \\ 3 & 2 & 1 \end{bmatrix} = C \begin{bmatrix} ? \\ ? \\ ? \end{bmatrix}$$

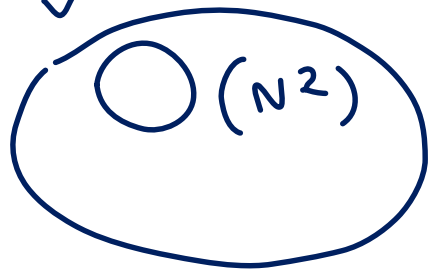


# Масиви

Сортиране по метод на мекъртею

Метод bubble sort

Сложност



Задача (10\*) Решете. [http://dev.bfu.bg/Spoj/contest\\$pl?contestid=48](http://dev.bfu.bg/Spoj/contest$pl?contestid=48)  
Задача #347 „SHOPPING маниа“ по метода на мекъртею

Задачи.

(8) Сортирайте едномерен масив с 10 елемента по метод на мекъртею.

Вход.

7	3	2	5	9	1	8	0	4	6
---	---	---	---	---	---	---	---	---	---

Изход.

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

(9) Сортирайте двумерен масив (3x3) по метод на мекъртею

Вход

9	1	7
4	6	2
8	3	5

Изход

1	2	3
4	5	6
7	8	9