

Динамична Памет

1)

Напишете функция map, която приема указател към масив, размера му и указател към функция int -> int. Функцията да прилага подадената функция върху всеки елемент в масива и да връща нов масив, съдържащ новите стойности. Всички масиви да са динамично заделени!

2)

Напишете функция flatmap, която приема указател към масив, размера му и указател към функция f. Функцията f приема int, указател към памет, достатъчна за $10 \times \text{int}$ и връща число. Считаме, че f генерира $k \leq 10$ елемента в подадената му памет. Функцията да връща указател към динамично заделен масив, съдържащ всички елементи, генериирани от f за всички елементи във първоначалния масив.

```
```cpp
int f(int n, int *arr) {
 int res = min(n, 10);
 for(int i = 0; i < res; ++i)
 arr[i] = n;
 return res;
}
...
int a[] = {1,2,3,4, 11};
flatmap(a, 4, f);
```

```

Изход: `1 2 2 3 3 3 4 4 4 4 11 11 11 11 11 11 11 11`

3)

Създайте функция, която създава матрица $n \times m$. Създайте функция умножаване на матрици (приема указатели към две матрици и размерите им и връща новозаделена матрица).

Напишете програма, която прочита две матрици с произволни размери от стандартния вход и принтира резултата от умножението им на стандартния изход.

4)

Създайте функция, която създава масив от n масива (n е параметър). Всеки от масивите съдържа пореде ред от редицата [look and say](#). Първият ред съдържа само едно число, което може да не е 1 (подава се като аргумент на функцията)

Изведете полученият масив на екрана.

Пример за първи елемент 2:

```
2
1 2
1 1 1 2
3 1 1 2
1 3 2 1 1 2
```

