

Задача 1. Задачата да се реши на езика C++.

Даден е двумерен масив с размер 6 на 6 от символи — малки и главни латински букви и цифри. Две клетки в него ще наричаме “съседни”, ако имат обща стена (т.е. всяка клетка е съседна с най-много четири други, намиращи се под, над, вляво и вдясно от нея). Път с дължина N ще наричаме редица a_0, a_1, \dots, a_N — от клетки, за която:

1. за всяко $0 \leq i < N - 1$ е изпълнено, че a_i и a_{i+1} са съседни;
2. никоя от клетките не се среща повече от веднъж (т.е. няма цикли).

Да се попълнят празните места в кода на дадените по-долу функция `contains` и помощната ѝ функция `walk`. Функцията `contains` получава два аргумента — масив `arr` от дадения тип `char[6][6]` и символен низ `str`. Тя трябва да връща истина тогава и само тогава, когато в `arr` съществува път, чиито клетки образуват точно съдържанието на низа `str` (вижте примера по-долу). За определеност считаме, че функцията трябва да връща истина за празния низ.

Пример: За дадения по-долу двумерен масив `contains` трябва да върне истина, ако ѝ бъдат подадени низовете `"abcdefgh"`, `"A123B123C"` или `""`. За улеснение, за да може да ги видите по-лесно, те са маркирани в сиво.

y	u	f	a	b	c
G	o	p	g	B	1
c	b	a	h	3	2
d	k	j	i	2	3
e	f	Q	N	1	C
h	g	h	M	A	r

Кодът на двете функции е даден на следващия лист:



```
bool contains(char arr[6][6], const char* str)
{
```

```
    for (int row = 0; row < ____; ____)  
        for (int col = 0; col < ____; ____)  
            if (walk(arr, row, col, str))  
                return ____;  
    return ____;  
}  
  
bool walk(char arr[6][6], int row, int col, const char* str)  
{  
    if (*str == '\\0')  
        return ____;  
  
    if (row < 0 || col < 0 || row >= 6 || col >= 6)  
        return ____;  
  
    if (arr[row][col] != *str)  
        return ____;  
  
    arr[row][col] *= -1;  
  
    bool result =  
        walk(arr, row + ____, col, str + 1) ||  
        walk(arr, ____, ____, str + 1) ||  
        walk(arr, ____, ____, str + 1) ||  
        walk(arr, ____, ____, str + 1);  
  
    arr[row][col] ____;  
  
    return result;  
}
```