

Сдать задание до:

3 октября (включительно)

Куда отправлять:

dima@pkims.ru

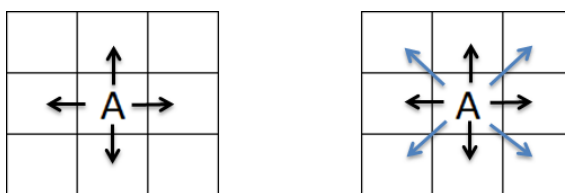
Тема письма:

ТА – СРС2 – группа – фамилия

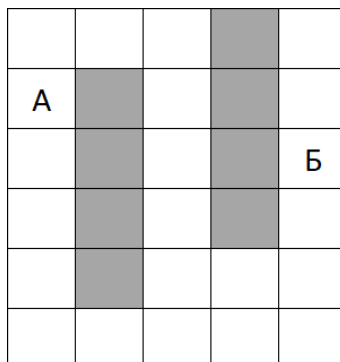
Простое задание

2 балла максимум, кодирование на C++ не требуется

Волновой алгоритм, с которым вы столкнулись в прошлом задании, в реальных играх применяется с другими правилами: для возможности плавного перемещения персонажа у каждой клетки рассматриваются не 4 соседа (пример на рисунке ниже слева), а с 8 (пример на рисунке ниже справа).



Вам дана матрица, на которой расставлены начальная точка «А» и конечная точка «Б» волнового алгоритма, а также расставлены препятствия (см. рисунок).



Из точки А можно выйти в две разные стороны (вверх и вниз). Напишите:

1. какое число шагов потребуется для достижения точки Б из точки А для обоих вариантов построения пути;
2. какое число шагов потребуется для достижения точки Б из точки А, если изменится число клеток для обхода с 4 до 8, как показано на первом рисунке справа.

Сложное задание

3 балла максимум, кодирования на C++ не требуется

Пару лет назад студенту было дано задание написать программу, которая должна пронормировать на 1 массив из 5 положительных ненулевых целых чисел (пронормировать на 1 – значит сделать так, чтобы максимальное число массива стало равным 1, а все остальные пропорционально уменьшились; например, если есть числа 10, 5 и 2, нормировка на 1 даст следующий результат: 1, 0.5, 0.2).

Мне был представлен результат, сокращённый код которого выглядит следующим образом:

```
01 void main()
02 {
03     float mas[5];
04
05     // Тут идёт некоторая инициализация массива, не принципиально
06
07     // А вот тут начинается нормировка
08     float max = 0;
09     for(int i = 0; i < 5; i+=1)
10     {
11         if(mas[i] > max)
12             max = mas[i];
13
14         mas[i] = mas[i] / max;
15     }
16
17     return 0;
18 }
```

Что в данном решении мне не понравилось? Напишите номер строки/номера строк, в которых по вашему мнению написан код, из-за которого нормировка работает неверно.

По вариантам восстановите последовательность чисел (вычислите X и Y и запишите исходную последовательность целиком), если известна часть последовательности и то, какой результат был получен после неправильной нормировки.

Запишите, как для этих чисел будет выглядеть ответ, если реализовать алгоритм правильно.

Номер варианта	Вариант задания
1, 15	Известная часть последовательности: 4.0 ; X ; 8.0 ; Y ; 2.0 Полученный результат: 1.0 ; 0.5 ; 1.0 ; 0.75 ; 0.25
2, 16	Известная часть последовательности: X ; Y ; 3.0 ; 5.0 ; 2.0 Полученный результат: 1.0 ; 1.0 ; 1.0 ; 1.0 ; 0.4
3, 17	Известная часть последовательности: 8.0 ; X ; Y ; 3.0 ; 5.0 Полученный результат: 1.0 ; 0.5 ; 1.0 ; 0.3 ; 0.5
4, 18	Известная часть последовательности: 5.0 ; X ; Y ; 3.0 ; 6.0 Полученный результат: 1.0 ; 0.8 ; 1.0 ; 0.5 ; 1.0
5, 19	Известная часть последовательности: X ; 2.0 ; 8.0 ; Y ; 8.0 Полученный результат: 1.0 ; 1.0 ; 1.0 ; 0.125 ; 1.0

6, 20	Известная часть последовательности: 4.0 ; X ; 8.0 ; Y ; 2.0 Полученный результат: 1.0 ; 0.5 ; 1.0 ; 0.75 ; 0.25
7, 21	Известная часть последовательности: X ; Y ; 3.0 ; 5.0 ; 2.0 Полученный результат: 1.0 ; 1.0 ; 1.0 ; 1.0 ; 0.4
8, 22	Известная часть последовательности: X ; 2.0 ; 8.0 ; Y ; 8.0 Полученный результат: 1.0 ; 1.0 ; 1.0 ; 0.125 ; 1.0
9, 23	Известная часть последовательности: 5.0 ; X ; Y ; 3.0 ; 6.0 Полученный результат: 1.0 ; 0.8 ; 1.0 ; 0.5 ; 1.0
10, 24	Известная часть последовательности: 2.0 ; X ; Y ; 5.0 ; 8.0 Полученный результат: 1.0 ; 1.0 ; 0.1 ; 0.5 ; 0.8
11, 25	Известная часть последовательности: X ; Y ; 3.0 ; 5.0 ; 2.0 Полученный результат: 1.0 ; 1.0 ; 1.0 ; 1.0 ; 0.4
12, 26	Известная часть последовательности: 2.0 ; X ; Y ; 5.0 ; 8.0 Полученный результат: 1.0 ; 1.0 ; 0.1 ; 0.5 ; 0.8
13, 27	Известная часть последовательности: 8.0 ; X ; Y ; 3.0 ; 5.0 Полученный результат: 1.0 ; 0.5 ; 1.0 ; 0.3 ; 0.5
14, 28	Известная часть последовательности: 8.0 ; X ; Y ; 3.0 ; 5.0 Полученный результат: 1.0 ; 0.5 ; 1.0 ; 0.3 ; 0.5