

## Задачи по информатике.

### Задача 1

- В1.** Решить уравнение в системе счисления  $113_x + 323_x = 441_x$
- В2.** Решить уравнение в системе счисления  $443_x + 323_x = 1066_x$
- В3.** Решить уравнение в системе счисления  $235_x + 345_x = 613_x$
- В4.** Решить уравнение в системе счисления  $414_x + 214_x = 631_x$
- В5.** Решить уравнение в системе счисления  $606_x = 259_x + 369_x$
- В6.** Решить уравнение в системе счисления  $220_x + 242_x = 570_x$

### Задача 2

**В1.** В электронном табло «бегущая строка» длина строки табло составляет 10 знакомест, символы, отображаемые в знакоместах, сдвигаются на 1 влево каждые 2 секунды. Исходно табло пустое. Начальным моментом показа будем считать момент появления первого символа на крайнем правом знакоместе. По прошествии двух секунд показа, этот символ сдвинется на одно знакоместо влево, а на его месте отобразится второй символ сообщения. Затем, каждые две секунды имеющиеся символы сдвигаются на одно знакоместо влево, а на освободившемся крайнем правом знакоместе появляется новый символ, пока сообщение не закончилось. Если сообщение закончилось, то при очередном сдвиге, крайнее правое знакоместо становится пустым, через 2 секунды пустыми окажутся два знакоместа в конце табло и так далее, пока все табло не станет пустым. Этот момент (когда табло опустело) будем считать завершением показа сообщения.

Символы, отображаемые в бегущей строке, берутся из набора из 40 различных символов, каждый из которых может выводиться красным или зеленым цветом. Сообщение хранится в памяти как последовательность нулей и единиц: каждый цвет каждого символа из набора кодируется двоичным кодом. Все символы кодируют одинаковым и минимально возможным количеством бит.

Для хранения сообщения в памяти табло отведено  $X$  байт. Известно, что максимальная продолжительность показа сообщения с момента появления первого символа, до момента исчезновения последнего символа составляет 98 секунд. Другими словами, самое большое сообщение, которое может поместиться в память табло, будет показано за 98 секунд. Определить память табло в байтах.

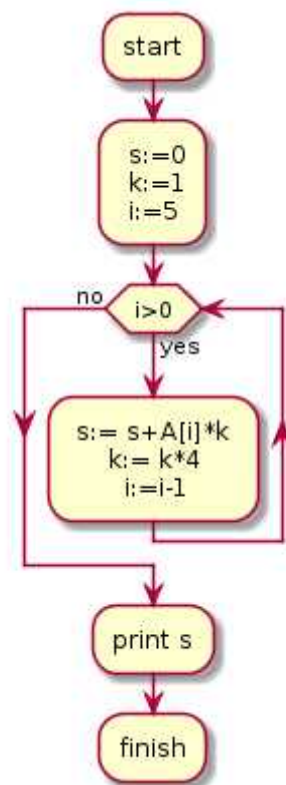
**В2.** В электронном табло «бегущая строка» длина строки табло составляет 10 знакомест, символы, отображаемые в знакоместах, сдвигаются на 1 влево каждые 2 секунды. Исходно табло пустое. Начальным моментом показа будем считать момент появления первого символа на крайнем правом знакоместе. По прошествии двух секунд показа, этот символ сдвинется на одно знакоместо влево, а на его месте отобразится второй символ сообщения. Затем, каждые две секунды имеющиеся символы сдвигаются на одно знакоместо влево, а на освободившемся крайнем правом знакоместе появляется новый символ, пока сообщение не закончилось. Если сообщение закончилось, то при очередном сдвиге, крайнее правое знакоместо становится пустым, через 2 секунды пустыми окажутся два знакоместа в конце табло и так далее, пока все табло не станет пустым. Этот момент (когда табло опустело) будем считать завершением показа сообщения.

Символы, отображаемые в бегущей строке, берутся из набора из 50 различных символов, каждый из которых может выводиться красным или зеленым цветом. Сообщение хранится в памяти как последовательность нулей и единиц: каждый цвет каждого символа из набора кодируется двоичным кодом. Все символы кодируют одинаковым и минимально возможным количеством бит.

Для хранения сообщения в памяти табло отведено  $X$  байт. Известно, что максимальная продолжительность показа сообщения с момента появления первого символа, до момента исчезновения последнего символа составляет 146 секунд. Другими словами, самое большое сообщение, которое может поместиться в память табло, будет показано за 146 секунд. Определить память табло в байтах.

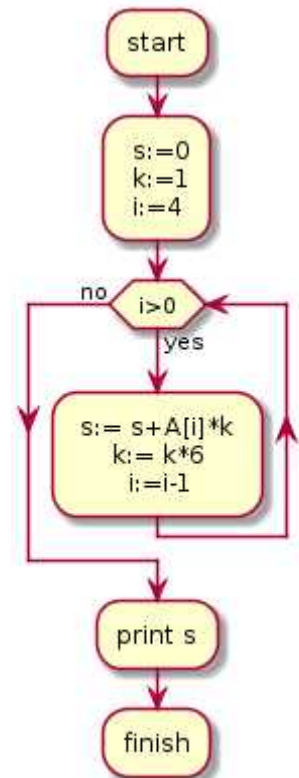
### Задача 3

**В1.** На рисунке блок-схема алгоритма. На вход алгоритму был подан массив  $A$  из пяти целых положительных чисел, таких что  $0 \leq A[i] \leq 3$  для всех  $i$ . Нумерация элементов массива начинается с 1. Найдите значения элементов этого массива, если известно, что после выполнения алгоритма получилось значение переменной  $s = 935$ . В ответе укажите через значения элементов массива в порядке возрастания индексов.



### Задача 3

**В2.** На рисунке блок-схема алгоритма. На вход алгоритму был подан массив **A** из четырех целых положительных чисел, таких что  $0 \leq A[i] \leq 5$  для всех  $i$ . Нумерация элементов массива начинается с **1**. Найдите значения элементов этого массива, если известно, что после выполнения алгоритма получилось значение переменной **s=321**. В ответе укажите через значения элементов массива в порядке возрастания индексов.



## Задача 4.

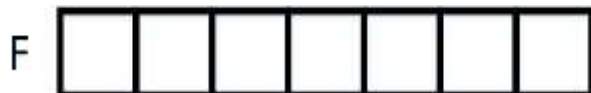
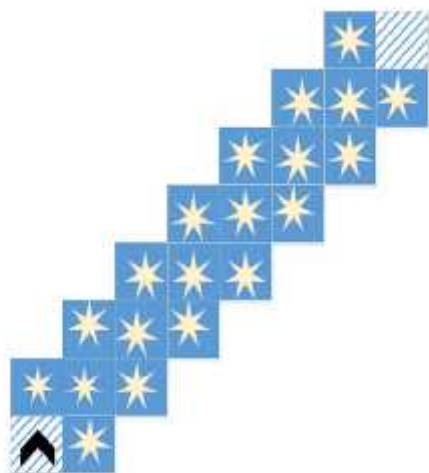
**В1.** Роботу-стрелке необходимо собрать все звездочки. Роботу-стрелке доступны 4 команды:

- команды поворота на месте налево или направо,
- команда одного шага вперед;
- команда вызова функции  $F$ .

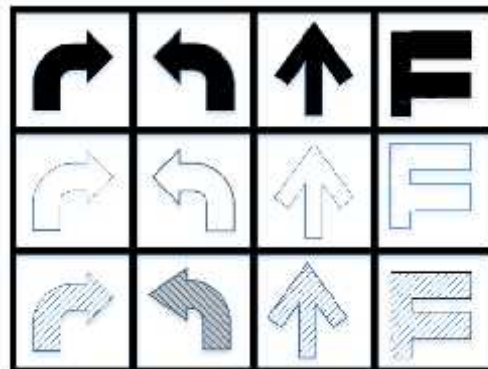
Кроме этого указанные команды могут выполняться на любой клетке, на заштрихованной или на прозрачной. Черные команды выполняются на любой клетке. Прозрачные и заштрихованные команды могут быть выполнены только на клетках соответствующего цвета.

Для написания программы для робота-стрелки доступно 7 ячеек памяти в функции  $F$ . Команды можно рекурсивно повторять, если в качестве команды вызвать саму функцию  $F$ .

Поле приведено на *рис. 1*



Доступные команды приведены на *рис. 2*.



### Задача 5.

**В2.** Роботу-стрелке необходимо собрать все звездочки. Роботу-стрелке доступны 4 команды:

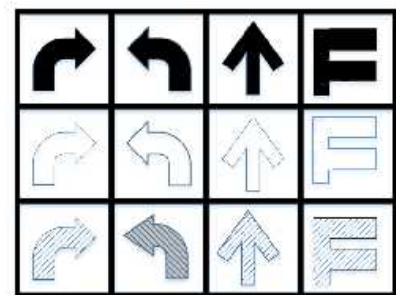
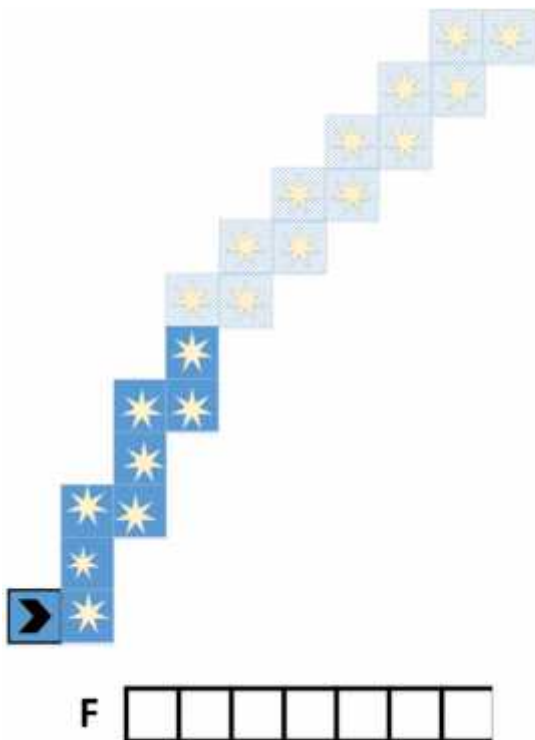
- команды поворота на месте налево или направо,
- команда одного шага вперед;
- команда вызова функции *F*.

Кроме этого указанные команды могут выполняться на любой клетке, на заштрихованной или на прозрачной. Черные команды выполняются на любой клетке. Прозрачные и заштрихованные команды могут быть выполнены только на клетках соответствующего цвета.

Для написания программы для робота-стрелки доступно 7 ячеек памяти в функции *F*. Команды можно рекурсивно повторять, если в качестве команды вызвать саму функцию *F*.

Поле приведено на рис. 1

Доступные команды приведены на рис. 2.



## Задача 5.

**В1.** Написать программу определяющую является ли многоугольник выпуклым по заданным координатам вершин.

**Входные данные:**

Первая строка – количество вершин многоугольника  $n$ . Последующие  $n$  строк – пары координат вершин  $(x, y)$ .

**Выходные данные:**

В случае если многоугольник является выпуклым вывести **1**, в противном случае **0**. Если фигура не является многоугольником вывести **-1**.

**Ограничения:** Значения координат являются целочисленными

**Пример:**

**Ввод:**

5  
86 384  
241 389  
339 245  
123 124  
143 268

**Вывод:**

0

## Задача 5.

**В2.** Заданы координаты вершин невыпуклого многоугольника. Необходимо найти такую вершину данного многоугольника, после перемещения которой на минимальное возможное расстояние многоугольник станет выпуклым.

### Входные данные:

Первая строка – количество вершин многоугольника  $n$ . Последующие  $n$  строк – пары координат вершин  $(x, y)$ .

### Выходные данные:

Вывести порядковый номер перемещаемой точки. Отсчет порядковых номеров начинается с **1**, первой точкой считается та, которая первой была подана на ввод.

**Ограничения:** Значения координат являются целочисленными

**Пример:**

**Ввод:**

5  
86 384  
241 389  
339 245  
123 124  
143 268

**Вывод:**

5

## Задача 5.

**В3.** Найдите в строке  $S$  длины  $N$  количество таких подстрок во фрагменте  $S[A, B]$ , в которых содержащиеся подстроки содержат не более  $K$  нулей и  $K$  единиц.

**Входные данные:**

Первая строка:  $N, K$

Вторая строка:  $S$ , содержащая только нули и единицы

Третья строка:  $A, B$

**Выходные данные:**

Целое число подстрок.

**Пример:**

**Ввод:**

8 2

01110000

1 4

**Вывод:**

8



### Задача 5.

**В4.** Числа  $A$  и  $B$  выбираются случайным образом на промежутке  $[1, \dots, N]$ . Найдите вероятность того, что наибольший общий делитель  $A$  и  $B$  будет равен  $B$ .

**Входные данные:**

Число:  $N$

:

Вероятность, округленная до 5 знака после запятой.

**3**

**0.55556**

## Задача 5.

**В5.** С клавиатуры вводится последовательность натуральных чисел (не превосходящих 10000), по одному числу в строке. Количество чисел заранее не известно, однако не превышает 10000. Признаком окончания последовательности является ввод числа «0», при этом само число 0 не является частью входной последовательности.

Искомыми подпоследовательностями входной последовательности назовем такие подпоследовательности, которые удовлетворяют двум условиям:

- состоят только из чисел, все цифры которых делятся на четыре (0 считается делящимся на любое число);
- являются неубывающими (каждое последующее число больше либо равно предыдущему).

Вывести на экран максимальную длину искомой подпоследовательности. Если искомым последовательностей нет, то вывести число 0.

8 2 40 4 48 84 2 804 0	3
2	2
4 16 88 88 88 88 0	4

## Задача 5.

**В6.** На вход программе подаётся последовательность целых чисел. В первой строке находятся два числа  $M$  и  $N$  ( $2 \leq M, N \leq 100$ ). В следующих  $N$  строках перечислено по  $M$  целых чисел (встречаются числа от 1 до 20). Необходимо найти все встречающиеся в таблице «плюсы». «Плюсом» считается набор из 5 равных друг другу соседних чисел: «центральное» число, и четыре вокруг него (сверху, слева, снизу и справа),

например:

```
1 5 0
5 5 5
1 5 3
```

Необходимо вывести на экран число встретившихся «плюсов». Если не удалось найти ни одного «плюса», то напечатать 0.

4 5 1 0 0 0 0 0 0 1 1 0 1 1 1 1 1 1 0 0 1 1	2
2	2
4 5 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 3 2 0 1 1	4

## Задача 5.

**В7.** На вход программе подаётся последовательность целых чисел. В первой строке находятся два числа  $M$  и  $N$  ( $2 \leq M, N \leq 100$ ). В следующих  $N$  строках перечислено по  $M$  целых чисел (встречаются числа от 1 до 20). Необходимо найти все встречающиеся в таблице «буквы Н». «Буквой Н» считается набор из 7 равных друг другу соседних чисел в форме буквы Н, например:

```
5 0 5
5 5 5
5 2 5
```

Необходимо вывести на экран число встретившихся «букв Н». Если не удалось найти ни одной — напечатать 0.

4 5 1 0 1 4 1 1 1 1 1 0 1 0 1 0 0 0 0 0 3 0	2
2	2
4 5 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 1 0	4

## Задача 6.

**В1.** Гоша делал презентацию по литературе и решил вставить в неё стихотворение. Но обычные стихотворения показались ему слишком скучными, поэтому он выбрал для своей презентации стих, в котором первые буквы строчек шли в алфавитном порядке подряд. Когда мальчик скопировал и вставил произведение в свою презентацию, в работе компьютера произошёл сбой, и все стихотворение оказалось напечатанным на слайде в одну строчку, а между буквами в словах появились знаки препинания, знаки математических действий, скобки. Помогите Гоше написать программу, которая поможет ему восстановить стихотворение в первоначальном виде, если известно, что в нем нет ни одного имени собственного.

- Программа должна быть написана на одном из следующих языков программирования: Basic, Pascal, C, C++, Алгоритмический. Другие языки программирования не допускаются.

**В2.** Вася писал сочинение и отошел попить чай на кухню. В это время его младший брат Миша вставил в текст математические выражения с числами и операциями сложения и вычитания, поскольку ему срочно нужно было решать контрольную по математике. Вася добрый брат и поэтому он решил написать программу, которая вычислит значения написанных братом выражений, но не смог. Помогите Васе написать программу, которая выполнит математические операции, не взирая на остальной текст, и посчитает результаты вычислений.

- Программа должна быть написана на одном из следующих языков программирования: Basic, Pascal, C, C++, Алгоритмический. Другие языки программирования не допускаются.