**Задания для 7-8 класса школьного этапа Всероссийской олимпиады по информатике.**

Каждая задача оценивается в 100 баллов.

**Задача 1. Робот**

Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости:

**вверх вниз влево вправо.**

При выполнении любой из этих команд РОБОТ перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →. Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

**сверху свободно снизу свободно**

**слева свободно справа свободно**

 Цикл **ПОКА <условие> команда** выполняется, пока условие истинно, иначе происходит переход на следующую строку. Сколько клеток приведенного лабиринта соответствуют требованию, что, выполнив предложенную ниже программу, РОБОТ уцелеет (не врежется в стену) и остановится в той же клетке, с которой он начал движение? В ответе укажите количество клеток, удовлетворяющих условию, и обозначьте знаком Х эти клетки.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**НАЧАЛО**

**ПОКА <слева свободно> вверх**

**ПОКА <сверху свободно> вправо**

**ПОКА <справа свободно> вниз**

**ПОКА <снизу свободно> влево**

**КОНЕЦ**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

***Решение***

Замечание: если указана 1 клетка – 50 баллов. Если лишние клетки – 0 баллов.

**Ответ: 2 клетки**

## Задача 2. Божьи коровки

### *Условие*

В теплый осенний день с 10:00 стена дома по адресу ул. Пушкина 35, прогревается под солнечными лучами, что привлекает божьих коровок.

Известно, что к 10:00 прилетают первая божья коровка и затем каждые 30 минут к каждой божьей коровке прилетают две ее подружки. Сколько божьих коровок будет через N минут?

Считать, что стена остается прогретой до 18:00.

### *Формат* входных данных

### В первой строке ввода содержится единственное число N – количество минут, прошедшее после прилета первой божьей коровки.

### *Формат выходных* данных

Выведите единственное целое число K – количество божьих коровок через N минут.

### *Ограничения*

N ограниченно от 0 до 480.

### *Примеры тестов*

|  |  |  |
| --- | --- | --- |
| **№** | **Входные данные** | **Выход** |
| 1 | 60 | 9 |

***Решение***

Решение задачи основывается на подсчете с использованием циклов либо рекурсии.

Возможные варианты решения задачи на языке Pascal.ABC:

С рекурсией:

**var**

s:longint;

 n: Integer = 0;

 **function** kor( n:integer):longint;

 **begin**

 **if** n>0 **then** kor:= kor(n-1)\*3 **else** kor:=1;

 **end**;

**begin**

 Readln(n);

 n:=n **div** 30;

 s:=kor(n);

 Write(s);

**end**.

Без рекурсии:

**var**

s:longint;

 n,i: Integer;

**begin**

 Readln(n);

 s:=1;

 n:=n **div** 30;

 **for** i:=1 **to** n **do**

s:=s\*3;

 Write(s);

**end**.

Возможный вариант решения задачи на языке C++:

#include <bits/stdc++.h>

using namespace std;

int N;

int ans = 1;

int main(int argc, char\*\* argv) {

 cin >> N;

 int halfes = N / 30;

 for (int i = 1; i <= halfes; ++i) {

 ans = ans \* 3;

 }

 cout << ans << '**\n**';

 return 0;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Вход** | **Выход** | **Баллы**  |
| 1 | 0 | 1 | 10 |
| 2 | 38 | 3 | 10 |
| 3 | 53 | 3 | 10 |
| 4 | 60 | 9 | 10 |
| 5 | 79 | 9 | 10 |
| 6 | 125 | 81 | 10 |
| 7 | 234 | 2187 | 10 |
| 8 | 345 | 177147 | 10 |
| 9 | 400 | 1594323 | 10 |
| 10 | 480 | 43046721 | 10 |
| **Итого** | **100** |

## Задача 3. Секретный замок

### *Условие*

На вход автомату подаются два натуральных числа не превышающих 10000000. Автомат сравнивает цифры на одинаковых позициях чисел. Выводит одно целое число – количество совпавших цифр. (Примечание: сравниваются по порядку цифры **с конца** числа).

Например, на вход автомату подаются числа 10322 и 302, на выходе автомат выведет число 2. Совпадают цифры 3 (3-я позиция с конца) и 2 (1-я позиция с конца).

### *Формат* входных данных

В первой строке ввода содержится два целых числа A, B.

### *Формат выходных* данных

Выведите единственное целое число K – количество совпавших цифр в числах.

### *Ограничения*

0 < A, B < 107.

### *Примеры тестов*

|  |  |  |
| --- | --- | --- |
| **№** | **Входные данные** | **Выход** |
| 1 | 3 6  | 0 |
| 2 | 10372 302 | 2 |

***Решение***

Решение основано на знании разрядов числа и умении выделять разряд из числа.

Возможный вариант решения задачи на языке Pascal.ABC (для поточного ввода данных):

**var**

n,m,kol\_cifr,min:longint;

**begin**

Readln(m,n);

 kol\_cifr:=0;

 **if** m>n **then** min:=n

**else** min:=m;

 **while** min<>0 **do**

 **begin**

 **if** m **mod** 10=n **mod** 10

**then** kol\_cifr+=1;

 m:=m **div** 10;

n:= n **div** 10;

min:=min **div** 10;

 **end**;

 Write(kol\_cifr);

**end**.

Возможный вариант решения задачи на языке C++:

#include <bits/stdc++.h>

using namespace std;

int A, B, ans;

int main(int argc, char\*\* argv) {

 cin >> A >> B;

 while (A && B) {

 int a = A % 10;

 int b = B % 10;

 if (a == b)

 ++ans;

 A = A / 10;

 B = B / 10;

 }

 cout << ans << '**\n**';

 return 0;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Вход** | **Выход** | **Баллы**  |
| 1 | 3 6  | 0 | 10 |
| 2 | 4 4 | 1 | 10 |
| 3 | 13 153 | 1 | 10 |
| 4 | 10372 302 | 2 | 10 |
| 5 | 100000 1 | 0 | 10 |
| 6 | 1 100000 | 0 | 10 |
| 7 | 567 765 | 1 | 10 |
| 8 | 4559898 4659897 | 5 | 10 |
| 9 | 123456 654321 | 0 | 10 |
| 10 | 9999999 9999999 | 7 | 10 |
| **Итого** | **100** |

**Задача 4. Сторож и лампочки**

***Условие***

Складское помещение некоторого предприятия имеет форму прямоугольника, разделенного на одинаковые квадратные сектора. Всего секторов на складе M×N. Вход на склад расположен в северо-западном углу здания. Сторож за ночь делает обход всего склада, проходя каждый сектор, двигаясь по часовой стрелке от сектора к сектору по спирали. **Заканчивается обход**, когда пройдены **все сектора**. Как только сторож поворачивает за угол, загорается лампочка.

Требуется написать программу, которая для заданных M и N, определяет сколько лампочек зажжётся во время обхода.

### *Формат* входных данных

В первой строке ввода содержится два целых числа M и N.

### *Формат выходных* данных

Требуется вывести единственное целое число K – количество лампочек.

***Ограничения***

1 ≤ M, N ≤ 100

***Примеры тестов***

|  |  |  |
| --- | --- | --- |
| **№** | **Входные данные** | **Выход** |
| 1 | 3 4 | 5 |

Изображённый ниже рисунок соответствует первому тестовому примеру.

***Решение***Сторож перемещается по заданной условием траектории (см. схему 1).

Т.к. секторы являются единичными квадратами прямоугольника размерностью M x N, то определяем наименьшее значение из измерений прямоугольника и увеличиваем в 2 раза, т.к. поворот сторож делает в начале и конце линии секторов. Исключение составляют первая линия и последняя. Проходя первую линию секторов, сторож поворачивает только один раз, что видно из схемы. Если длина склада меньше ширины (M<N), то только в первой линии сторож поворачивает один раз и тогда количество включившихся лампочек будет равно M\*2-1, иначе в последней линии прохода тоже будет только 1 поворот и тогда количество загоревшихся лампочек будет равно M\*2-2.

Схема 1

Возможный вариант решения задачи на языке Pascal.ABC:

**var**

n,m,kol\_lamp:longint;

**begin**

 Readln(m,n);

 **if** m<n **then** kol\_lamp:=m\*2-1 **else** kol\_lamp:=n\*2-2;

 Write(kol\_lamp);

**end**.

Возможный вариант решения задачи на языке C++:

#include <bits/stdc++.h>

using namespace std;

int M, N, ans;

int main(int argc, char\*\* argv) {

 cin >> M >> N;

 ans = min(N - 1, M - 1) \* 2 + (N > M);

 cout << ans << '**\n**';

 return 0;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Вход** | **Выход** | **Баллы**  |
| 1 | 1 1 | 0 | 10 |
| 2 | 4 3 | 4 | 10 |
| 3 | 3 4 | 5 | 10 |
| 4 | 6 4 | 6 | 10 |
| 5 | 5 5 | 8 | 10 |
| 6 | 20 30 | 39 | 10 |
| 7 | 40 40 | 78 | 10 |
| 8 | 40 50 | 79 | 10 |
| 9 | 50 100 | 99 | 10 |
| 10 | 100 100 | 198 | 10 |
| **Итого** | **100** |

## Задача 5. Гимнасты

### *Условие*

На краевые соревнования по гимнастике отбирают спортсменов. Допустимый вес одного спортсмена варьируется в районе от 45 до 55 кг.

Определите, какое максимальное количество спортсменов из N спортсменов может поехать на соревнования, если максимальная грузоподъемность автобуса M кг.

Пример: всего на соревнования представлено 5 спортсменов, грузоподъемность автобуса 110 кг. Веса спортсменов: 45 56 62 55 49. Ответ: поедет 2 спортсмена (например, спортсмены с весами 45 и 55 кг).

### *Формат входного файла*

Первая строка ввода содержит натуральное число N – количество спортсменов, вторая строка ввода содержит натуральное число M – грузоподъемность автобуса, третья строка ввода содержит N натуральных чисел – веса спортсменов.

### *Формат выходного файла*

Выведите единственное целое число K  – максимальное количество подходящих по весу спортсменов, которые могут поехать на соревнования.

### *Ограничения*

N <= 10

M <= 600

***Примеры тестов***

|  |  |  |
| --- | --- | --- |
| **№** | **Входные данные** | **Выход** |
| 1 | 511045 56 62 55 49 | 2 |
| 2 | 815065 47 43 55 67 61 56 45 | 3 |

***Решение***

Возможный вариант решения задачи на языке Pascal.ABC:

**var**

a: **array** [1..10] **of** integer;

 i, n, m, k, s, j, c: integer;

**begin**

readln(n);

 readln(m);

 k := 0;

 **for** i := 1 **to** n **do**

read(a[i]);

 **for** i := 1 **to** n - 1 **do**

 **for** j := 1 **to** n - 1 **do**

 **if** a[j] > a[j + 1] **then begin**

c := a[j]; a[j] := a[j + 1]; a[j + 1] := c;

 **end**;

 s := 0;

 **for** i := 1 **to** n **do**

 **begin**

 **if** (a[i] >= 45) **and** (a[i] <= 55) **and** (s + a[i] <= m) **then**

 **begin**

s := s + a[i];

 k := k + 1;

 **end**;

 **end**;

 writeln(k);

**end**.

Возможный вариант решения задачи на языке C++:

#include <bits/stdc++.h>

using namespace std;

int N, M, a[15], ans;

int main(int argc, char\*\* argv) {

 cin >> N >> M;

 for (int i = 1; i <= N; ++i) {

 cin >> a[i];

 }

 sort(a + 1, a + 1 + N);

 for (int i = 1; i <= M; ++i) {

 if (a[i] >= 45 && a[i] <= 55 && a[i] <= M) {

 ++ans;

 M -= a[i];

 }

 }

 cout << ans << '**\n**';

 return 0;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Вход** | **Выход** | **Баллы**  |
| 1 | 715023 67 76 56 57 43 41 | 0 | 10 |
| 2 | 815065 47 43 55 67 61 56 45 | 3 | 10 |
| 3 | 617545 47 45 57 56 49 | 3 | 10 |
| 4 | 25045 55 | 1 | 10 |
| 5 | 412055 55 55 55 | 2 | 10 |
| 6 | 620045 47 48 50 57 61 | 4 | 10 |
| 7 | 1040055 45 48 50 54 47 51 52 53 55 | 8 | 10 |
| 8 | 511045 56 62 55 49 | 2 | 10 |
| 9 | 830067 50 55 45 47 53 50 38 | 6 | 10 |
| 10 | 1055055 45 48 50 54 47 51 52 53 55 | 10 | 10 |
| **Итого** | **100** |