Министерство науки и высшего образования Российской Федерации

ФГАОУ ВО «УрФУ имени первого Президента России Б.Н.Ельцина»

ИРИТ-РТФ

Центр ускоренного обучения

**ОТЧЕТ ПО РЕШЕНИЮ ЗАДАЧ НА ПЛАТФОРМЕ TIMUS**

по дисциплине «Программирование»

**Тема:** «Решение задач на платформе timus»

Студент группы РИВ-110938у: Вахрушев Н.А.

Преподаватель: Архипов Н.А.

Екатеринбург 2021

Цель

Получить опыт решения задач на электронном ресурсе с возможностью автоматической проверки своего решения.

Описание работы:

В ходе работы необходимо было ознакомиться с электронным ресурсом timus и выполнить 7 любых задач.

В качестве инструмента для решения всех задач был выбран язык программирования высокого уровня C++. При проверке использовался компилятор G++ 9.2 x64.

Ниже представлены гиперссылки на все выполненные задачи:

[Задача №1](https://acm.timus.ru/problem.aspx?space=1&num=1409)

[Задача №2](https://acm.timus.ru/problem.aspx?space=1&num=1785)

[Задача №3](https://acm.timus.ru/problem.aspx?space=1&num=2012)

[Задача №4](https://acm.timus.ru/problem.aspx?space=1&num=1877)

[Задача №5](https://acm.timus.ru/problem.aspx?space=1&num=1264)

[Задача №6](https://acm.timus.ru/problem.aspx?space=1&num=1787)

[Задача №7](https://acm.timus.ru/problem.aspx?space=1&num=1263)

Ход работы

Задача №1

Бандиты Гарри и Ларри отдыхали на природе. Решив пострелять, они выставили на бревно несколько банок из-под пива (не больше 10). Гарри начал простреливать банки по порядку, начиная с самой левой, Ларри — с самой правой. В какой-то момент получилось так, что они одновременно прострелили одну и ту же последнюю банку.

Гарри возмутился и сказал, что Ларри должен ему кучу денег за то, что тот лишил его удовольствия прострелить несколько банок. В ответ Ларри сказал, что Гарри должен ему еще больше денег по тем же причинам. Они стали спорить кто кому сколько должен, но никто из них не помнил сколько банок было в начале, а искать простреленные банки по всей округе было неохота. Каждый из них помнил только, сколько банок прострелил он сам.

Определите по этим данным, сколько банок не прострелил Гарри и сколько банок не прострелил Ларри.

**Исходные данные**

В единственной строке записано 2 числа — количество банок, простреленных Гарри и Ларри соответственно.

**Результат**

Выведите 2 числа — количество банок, не простреленных Гарри и Ларри соответственно.

Пример:

|  |  |
| --- | --- |
| исходные данные | результат |
| 4 7 | 6 3 |

Исходный код:

#include <iostream>

int main()

{

int a{},b{},all{};

std::cin>>a>>b;

all=(a+b)-1;

std::cout<<all-a<<" "<<all-b<<std::endl;

}

Задача №2

Компания Lavin Interactive, разработчик пошаговой стратегии Losers-V, постоянно расширяет рынки сбыта и создаёт локализации своей игры даже на самые малоизвестные языки. В том числе, их заинтересовал язык племени австралийских аборигенов аниндилъяква.

Но в языке аниндилъяква нет числительных. Как же, например, перевести на него фразу «у вас семь чёрных драконов, а у вашего врага — сорок»? Локализаторы решили перевести её так: «у вас немного чёрных драконов, а у вашего врага — толпа». Они составили таблицу, в которой указали правила замены чисел, обозначающих количество монстров, на существительные и местоимения:

|  |  |  |
| --- | --- | --- |
| Количество | Обозначение на русском языке | Обозначение на языке аниндилъяква |
| от 1 до 4 | несколько | few |
| от 5 до 9 | немного | several |
| от 10 до 19 | отряд | pack |
| от 20 до 49 | толпа | lots |
| от 50 до 99 | орда | horde |
| от 100 до 249 | множество | throng |
| от 250 до 499 | сонмище | swarm |
| от 500 до 999 | полчище | zounds |
| от 1000 | легион | legion |

Помогите локализаторам автоматизировать процесс — напишите программу, которая по количеству монстров выдаст соответствующее этому количеству слово.

Исходные данные

В единственной строке записано целое число n (1 ≤ n ≤ 2000) — количество монстров.

Результат

Выведите слово, соответствующее данному количеству монстров на языке аниндилъяква.

Примеры:

|  |  |
| --- | --- |
| исходные данные | результат |
| 7 | several |
| 40 | lots |

Исходный код:

#include <iostream>

int main()

{

int number {};

std::cin>>number;

if(number>=1 && number<=4)std::cout<<"few"<<std::endl;

if(number>=5 && number<=9)std::cout<<"several"<<std::endl;

if(number>=10 && number<=19)std::cout<<"pack"<<std::endl;

if(number>=20 && number<=49)std::cout<<"lots"<<std::endl;

if(number>=50 && number<=99)std::cout<<"horde"<<std::endl;

if(number>=100 && number<=249)std::cout<<"throng"<<std::endl;

if(number>=250 && number<=499)std::cout<<"swarm"<<std::endl;

if(number>=500 && number<=999)std::cout<<"zounds"<<std::endl;

if(number>=1000)std::cout<<"legion"<<std::endl;

}

Задача №3

Гриша Н. сказал двум своим товарищам по команде, что даже если они не придут на четвертьфинал чемпионата мира, он решит на соревновании все предложенные задачи. Сокомандники Грише не поверили, и тогда он рассказал им план, как он собирается это сделать.

За первый час он хочет решить f задач. Если до конца первого часа ещё останется время, Гриша просто пойдёт гулять по коридору. Начиная со второго часа Гриша хочет тратить на каждую из оставшихся задач ровно по 45 минут. Если план Гриши удастся, сможет ли он решить за 5 часов соревнования все 12 предложенных задач?

Исходные данные

В единственной строке записано целое число f — сколько задач хочет решить Гриша за первый час соревнования (1 ≤ f ≤ 11).

Результат

Выведите «YES», если Грише в одиночку удастся решить все предложенные задачи, и «NO» в противном случае.

Примеры:

|  |  |
| --- | --- |
| Исходные данные | Результат |
| 7 | YES |
| 5 | NO |

Исходный код:

#include <iostream>

int main()

{

int number{};

std::cin>>number;

if((12-number)\*45<=4\*60)std::cout<<"YES"<<std::endl;

else std::cout<<"NO"<<std::endl;

}

Задача №4

У Дена есть два четырёхзначных кодовых замка для велосипеда. Каждый вечер он ставит велосипед на сигнализацию и пристёгивает к специальной стойке одним из замков. Ден никогда не использует один и тот же замок два вечера подряд. В некоторую ночь злоумышленник попытался с помощью кода 0000 открыть висящий на велосипеде замок. Сработала сигнализация, и вор поспешил скрыться. На следующую ночь он решил попробовать код 0001, затем 0002 и так далее в порядке возрастания номера.

Известно, что Ден не меняет кодов и в ночь, когда вор пришёл впервые, велосипед был пристёгнут первым замком.

Исходные данные

В первой строке записан код, установленный на первом замке, во второй строке — код, установленный на втором замке. Оба кода — строки длины 4, состоящие из цифр от 0 до 9.

Результат

Выведите «yes», если злоумышленник рано или поздно взломает замок, и «no» в противном случае.

Примеры:

|  |  |
| --- | --- |
| Исходные данные | Результат |
| 0001  0000 | no |
| 0002  0001 | yes |

Исходный код:

#include <iostream>

int main()

{

int first{},second{},bad\_man{0000};

std::cin>>first>>second;

while(1)

{

if(bad\_man==first){

std::cout<<"yes"<<std::endl;

break;

}

bad\_man++;

if(bad\_man==second)

{

std::cout<<"yes"<<std::endl;

break;

}

bad\_man++;

if(bad\_man>first && bad\_man>second)

{

std::cout<<"no"<<std::endl;

break;

}

}

}

Задача №5

После успеха предыдущей программы Васечкина, позволившей подвести итоги выборов всего за два дня, Васечкин был назначен начальником отдела. Не правда ли, успех? В данный момент Артемий Сидорович готовит техническое задание для своего подчиненного — программиста Петечкина. Задание заключается в написании крайне полезной функции, которая намного облегчит жизнь всем программистам отдела. Для каждого числа от 0 до M функция будет подсчитывать число раз, которое число встречается в N-элементном массиве. Артемий Сидорович полагает, что функция должна работать следующим образом (пример кода для N = 3, M = 1):

|  |  |
| --- | --- |
| C | Pascal |
| if (arr[0]==0) ++count[0];  if (arr[0]==1) ++count[1];  if (arr[1]==0) ++count[0];  if (arr[1]==1) ++count[1];  if (arr[2]==0) ++count[0];  if (arr[2]==1) ++count[1]; | if arr[0]=0 then count[0] := count[0] + 1;  if arr[0]=1 then count[1] := count[1] + 1;  if arr[1]=0 then count[0] := count[0] + 1;  if arr[1]=1 then count[1] := count[1] + 1;  if arr[2]=0 then count[0] := count[0] + 1;  if arr[2]=1 then count[1] := count[1] + 1; |

Артемий Сидорович хочет оценить время, за которое Петечкин справится с заданием. Известно, что Петечкин пишет одну строчку кода за одну секунду (не правда ли, очень быстро?). Артемий Сидорович точно не знает, какими M и N можно ограничиться. Ваша цель — написать программу, которая подсчитает число секунд, которые потребуется Петечкину на написание кода.

Исходные данные

Единственная строка содержит целые числа N (0 ≤ N ≤ 40000) и M (0 ≤ M ≤ 40000).

Результат

Выведите число секунд, требуемых Петечкину на написание функции.

Пример

|  |  |
| --- | --- |
| Исходные данные | Результат |
| 3 1 | 6 |

Исходный код:

#include <iostream>

int main()

{

int a{},b{},count{0};

std::cin>>a>>b;

for(int i=0;i<=b;i++)

for(int k=0;k<a;k++)

count++;

std::cout<<count<<std::endl;

}

Задача №6

Светофор, установленный на повороте в торговый центр «МЕГА» с Новомосковского тракта, работает в таком режиме, что за одну минуту повернуть успевают k машин. По выходным, когда жители города едут закупаться продуктами и одеждой, перед этим поворотом вырастает огромная пробка. Администрация торгового центра дала указание повесить под строящимся неподалёку мостом камеру, подсчитывающую количество автомобилей, подъезжающих к этому повороту со стороны города. По информации, полученной с камеры за n минут, прошедших с начала наблюдений, определите количество машин, стоящих в пробке в настоящий момент времени.

Исходные данные

В первой строке записаны целые числа k и n (1 ≤ k, n ≤ 100) — количество машин, успевающих повернуть на «МЕГУ» в течение минуты, и количество минут, прошедших с начала наблюдений. Во второй строке через пробел записаны целые числа a1, …, an (0 ≤ ai ≤ 100), где ai — количество машин, подъехавших к повороту со стороны города в течение i-й минуты. Можно считать, что наблюдения начинаются рано утром, когда машин, ожидающих на повороте, ещё нет.

Результат

Выведите количество машин, стоящих в пробке на повороте через n минут после начала наблюдений.

Примеры:

|  |  |
| --- | --- |
| Исходные данные | Результат |
| 5 3  6 7 2 | 0 |
| 5 3  20 0 0 | 5 |

Исходный код:

#include <iostream>

int main()

{

int number{},time{},count{0},problem{0},keep\_stay{0};

std::cin>>number>>time;

int array[time];

for(int i=0;i<time;i++)

{

std::cin>>array[i];

array[i]+=keep\_stay;

if(array[i]>number)array[i]=array[i]-number;

else array[i]=0;

keep\_stay=array[i];

}

std::cout<<keep\_stay<<std::endl;

}

Задача №7

Грядут очередные выборы. Снова все заборы оклеены листовками, почтовые ящики забиты макулатурой, с экранов телевизоров на нас взирают мордатые дядьки, обещающие сделать нашу жизнь лучше… А программист Васечкин снова завален работой. Необходимо написать программу, которая облегчит подсчет голосов избирателей.

Исходные данные

Первая строка содержит целые числа N — число кандидатов (1 ≤ N ≤ 10000) и M — число избирателей, принявших участие в выборах (1 ≤ M ≤ 10000). Далее следуют M строк, в каждой из которых находится номер кандидата, за которого проголосовал избиратель. Кандидаты пронумерованы целыми числами от 1 до N.

Результат

Выведите N строк, в i-й строке должен быть указан процент избирателей, проголосовавших за i-го кандидата (с точностью до двух знаков после десятичной точки).

Пример

|  |  |
| --- | --- |
| Исходные данные | Результат |
| 3 6  1  2  3  2  1  1 | 50.00%  33.33%  16.67% |

Исходный код:

#include <iostream>

#include <iomanip>

int main()

{

int a{},b{},cand{};

std::cin>>a>>b;

float ar1[a];

for(int i=0;i<a;i++)

ar1[i]=0.0;

int control=b;

while(b>0)

{

std::cin>>cand;

ar1[cand-1]+=1;

b--;

}

for(auto h:ar1)

std::cout<<std::fixed<<std::setprecision(2)<<(h\*100)/control<<"%"<<std::endl;

}

Вывод

В результате работы было произведено знакомство с электронным ресурсом timus и выполнены 7 задач, которые были проверены через внутренние инструменты сайта и отмечены, как решенные правильным способом.