

Московский государственный технический университет
имени Н.Э.Баумана

Олимпиада школьников «Шаг в будущее»

Инженерное дело «Профессор Лебедев» ИНФОРМАТИКА 2 тур

2018-2019 учебный год

10 класс

Вариант 2

Задача 1.

Андрей обожает рельсовый транспорт. В кармане он всегда носит свой любимый трамвай *Tatra T3*, игрушечный, конечно. В очередной раз он упросил маму не ехать на машине, а ехать на метро. В метро Андрей попросил маму подождать, пока он не научится предсказывать появление поезда. На часах было ровно h часов и m минут, когда показался первый поезд. Помогите Андрею рассчитать, в какой момент появится n -ый поезд, если интервал движения поездов составляет dm минут, ds секунд.

На вход программы подаются пять чисел: h , m , n и dm с ds , записанные через один пробел.

Программа должна вывести через пробел два целых числа без ведущих нулей: сколько часов и минут должно быть на часах метрополитена, исходя из имеющихся данных.

Пример

Входные данные	Выходные данные
18 00 4 2 30	18 7

Комментарий к примеру:

1-ый поезд – 18:00:00, 2-ой – 18:02:30, 3-ий – 18:05:00, 4-ый – **18:07:30**.

Решение

```
var h, m, n, dm, ds, t: longint;
```

```
Begin
```

```
  read(h, m, n, dm, ds);
```

```
  t := (h * 60 + m) * 60 + (n - 1) * (dm * 60 + ds);
```

```
  t := t div 60;
```

```
  write(t div 60 mod 24, ' ', t mod 60);
```

```
End.
```

Тесты

N	Оценка	Входные данные	Выходные данные
1	1	23 50 1 3 20	23 50
2	1	23 50 3 3 20	23 56
3	1	23 50 4 3 20	0 0

Задача 2.

Петя скоро пойдёт в школу, и он влюблён в числа. Недавно ему рассказали, как считать после девяти. Особенно Пете нравились числа, у которых нечётная сумма цифр. Он принялся их считать.

Напишите программу, которая проверит Петины подсчёты.

На вход программы подаётся два числа A и B , записанные через пробел ($1 \leq A \leq B \leq 10^9$).

Программа должна вывести количество чисел от A до B (включая сами числа A и B), которые так понравились Пете.

Пример

Входные данные	Выходные данные
20 30	6

Комментарий к примеру:

от 20 до 30 есть 6 искомым чисел: 21, 23, 25, 27, 29 и 30.

Решение

```
function nice(n: longint): boolean;
```

```
var s: longint;
```

```
begin
```

```
  s := 0;
```

```
  while n > 0 do begin
```

```
    inc(s, n mod 10);
```

```
    n := n div 10;
```

```
  end;
```

```
  nice := s mod 2 = 0;
```

```
end;
```

```
function count(a, b: longint): longint;
```

```
var res, i: longint;
```

```
begin
```

```
  res := 0;
```

```
  for i := a to b do
```

```

    if nice(i) then
        inc(res);
    count := res;
end;

var a, b, k, a1, b1: longint;
Begin
    read(a, b);
    k := 0;
    if b >= a then begin
        a1 := min(b, (a div 10 + 1) * 10 - 1); // конец текущего десятка A - aaa9
        inc(k, count(a, a1));
        a := a1 + 1;
        if b >= a then begin
            b1 := max(a, b div 10 * 10); // последний десяток bbb0 - xxxb
            inc(k, count(b1, b));
            b := b1 - 1;
            if b >= a then
                inc(k, 5 * (b + 1 - a) div 10);
        end;
    end;

    write(k)
End.

```

Тесты

N	Оценка	Входные данные	Выходные данные
1	1	99 99	0
2	2	99 300	101
3	2	999 3000	1002
4	2	999 1001	2
5	3	15 999999985	70503255

Задача 3.

Есть исполнитель *Сдвиг*. Он преобразует целое неотрицательное число, которое хранится в одном байте, по следующему правилу:

Если количество единиц в двоичной записи этого числа нечётное, то смещаем все двоичные цифры в байте на 2 позиции влево и записываем на образовавшиеся место два бита со значением *01*. Иначе число не изменяется. Например,

19_{10}

00010011_2

Преобразуется в

$010011_ _$ (сдвиг влево)

01001101_2 (результат)

77_{10}

Напишите программу, которая вычисляет наибольшее число, которое можно получить при преобразовании чисел из интервала $[A; B]$.

На вход программы подаётся два целых числа A и B , записанные через пробел.

Программа должна вывести одно целое число - вычисленное значение.

Пример

Ввод	Вывод
18 20	77

Решение

```
function f(n:integer):integer;
var s,x:integer;
begin
  x:=n;
  s:=0;
  while x>0 do
  begin
    s:=s+ x mod 2;
    x:= x div 2;
  end;
  if s mod 2 =0 then
    f:=n
  else
    f:=1+ (n * 4)mod 256
  end;

var a,b,i,x,max,imax:integer;
begin
  readln(a,b) ;
  for i:=a to b do
  begin
    x:=f(i);
    if (i=a)or (max<x) then begin
      max:=x;
      imax:=i
    end;
  end;
  writeln(max);
end.
```

Тесты

N	Оценка	Входные данные	Выходные данные
1	3	17 18	18
2	3	20 26	105
3	3	30 36	141
4	3	161 162	137
5	3	125 130	253

Задача 4.

Захват манипулятора способен выполнить три команды: «взять», «держат» и «выпустить». Поскольку захват разрабатывали студенты-троечники, он сделан кое-как и при попытке два раза подряд выполнить команду «взять» или «выпустить» ломается. В начальный момент времени захват выполнил команду «выпустить». Сколько можно составить разных последовательностей команд, длина которых не превышает заданное число n , так, чтобы захват не сломался?

Формат ввода

В строке вводится сначала натуральное число n – количество команд ($n \leq 15$).

Формат вывода

Вывести одно целое число – количество способов, которыми можно составить программы длиной не более n для захвата.

Пример

Входные данные	Выходные данные
2	7

Решение

```
program my;
var n,i,a,b,c,d,pa,pb,pc,sum:longint;
begin
  pa:=0;
  pb:=0;
  pc:=1;
  sum:=0;
  readln(n);
  for i:=1 to n do
  begin
```

```

a:=pb+pc;
b:=pa+pb+pc;
c:=pb+pa;
pa:=a;
pb:=b;
pc:=c;
sum:=sum+a+b+c;
end;
writeln(sum);
end.

```

Тесты

N	Оценка	Входные данные	Выходные данные
1	5	15	803759
2	5	3	19
3	5	1	2
4	5	10	9799
5	5	8	1680

Задача 5.

Суперагентам Матильде и Витгеру необходимо получить кодовый ключ к замку. Для того, чтобы составить ключ необходимо записать в одну строку без пробелов все слова максимальной возможной длины, составленные из букв некоторого алфавита, в которых каждая буква повторяется не более N раз ($N \leq 5$). Слова можно записать в любом порядке.

В качестве кода требуется набрать два числа: длину строки и количество слов в строке.

Входные данные:

На вход программе подаётся строка, содержащая алфавит, из которого надо составлять слова, в виде последовательности символов, записанных без пробела и число N , записанное через пробел после алфавита. В качестве буквы алфавита может быть использован любой символ. Количество букв в алфавите не превышает 5.

Выходные данные:

Два числа, записанные через один пробел. Первое число - количество слов, второе - длина строки.

Пример:

Входные данные	Выходные данные
<i>AB 2</i>	<i>6 24</i>

Пояснение:

Получится составить следующие 6 слов максимальной длины: ***AABB ABAB ABBA
BAAB BABA BBAА***

Решение

```
type ansistring = string;
function count_letter(const symbol, wrd: ansistring): longint;
var i, amount: longint;
begin
    amount := 0;
    for i := 1 to length(wrd) do
        if copy(wrd, i, 1) = symbol then
            amount := amount + 1;
    count_letter := amount;
end;
procedure count_and_build_words(N: longint; const alphabet: ansistring; const wrd: ansistring;
var amount, all_length: longint);
var i: longint;
    is_end: boolean;
begin
    is_end := true;
    for i := 1 to length(alphabet) do
        if count_letter(copy(alphabet, i, 1), wrd) < N then begin
            is_end := false;
            count_and_build_words(n, alphabet, wrd + copy(alphabet, i, 1), amount, all_length);
        end;
    if is_end then begin
        amount := amount + 1;
        all_length := all_length + length(wrd);
    end
end;
function fact(f: int64): int64;
var i, p: int64;
begin
    p := 1;
    for i := f downto 1 do
        p := p * i;
    fact := p;
end;
procedure count_and_build_words_fast(n: longint; const alphabet: ansistring; const wrd:
ansistring; var amount2, all_length2: longint);
var wrd_len, i: longint;
    is_end: boolean;
    amount, all_length: int64;
begin
```

```

wrд_len := length(alphabet) * n;
amount := 1;
all_length := wrд_len;
for i := 1 to length(alphabet) do begin
    amount := amount * fact(all_length) div fact(all_length - n) div fact(n);
    all_length := all_length - n;
end;
all_length := amount * wrд_len;
amount2 := amount;
all_length2 := all_length;
end;
var
    alphabet: ansistring;
    amount, all_length, n: longint;
begin
    read(alphabet);
    readln(n);
    amount := 0;
    all_length := 0;
    count_and_build_words_fast(n, alphabet, '', amount, all_length);
    writeln(amount, ' ', all_length);
end.

```

Тесты

N	Оценка	Входные данные	Выходные данные
1	3	AB 5	252 2520
2	5	ABC 2	90 540
3	5	ABCD 2	2520 20160
4	6	ABC 5	756756 11351340
5	8	ABCD 4	63063000 1009008000

Задача 6.

Коля хочет защитить файл от копирования и вводит пароль. Он принял решение использовать только десятичные цифры **0,1,2,3,4,5** и накладывает дополнительное условие. При посимвольном прочтении пароля количество четных цифр в последовательности уже прочитанных знаков никогда не превысит количество нечетных. При этом Коля почему-то не считает ноль четной цифрой, и поэтому просто игнорирует его при подсчете. Например, комбинация **012354** может быть паролем, а вот комбинация **211** – нет. Помогите Коле проверить надежность пароля и напишите программу, которая будет рассчитывать количество возможных паролей заданной длины.

Формат ввода

В строке вводится целое положительное число n – количество символов в пароле ($n \leq 10$).

Формат вывода

Вывести одно целое число – количество способов, которыми можно собрать пароль заданной длины, удовлетворяющий наложенным условиям. Если не существует ни одной такой комбинации, то вывести 0 .

Пример

Входные данные	Выходные данные
2	22

Решение

```
program my;
const N=10;
      M=N;

type tmatrix = array [0..N,0..M] of longint;

procedure init(var matrix:tmatrix);
var i,j:integer;
begin
  for i:=0 to N do
    for j:=0 to M do
      matrix[i,j]:=0;
    matrix[0,0]:=1;
  end;

function calc(matrix:tmatrix; kol:integer):longint;
var i,j:integer;
begin
  for i:=1 to kol do
    begin
      for j:=0 to M-1 do
        matrix[i,j]:=matrix[i,j]+matrix[i-1,j+1]*2;
      for j:=0 to M do
        matrix[i,j]:=matrix[i,j]+matrix[i-1,j];
      for j:=1 to M do
        matrix[i,j]:=matrix[i,j]+matrix[i-1,j-1]*3;
      end;
      j:=0;
    for i:=0 to M do
      j:=j+matrix[kol,i];
      calc:=j;
    end;
end;
```

```

var
  matrix:tmatrix;
  kol:integer;
begin
  readln(kol);
  init(matrix);
  writeln(calc(matrix,kol));
end.

```

Тесты

N	Оценка	Входные данные	Выходные данные
1	5	3	118
2	5	5	3802
3	3	1	4
4	7	6	21970
5	10	8	743494

Задача 7 (Ситуационная задача. Оценивается $\max = 30$ баллов).

Сердечник подкалиберного снаряда представляет из себя комбинацию тел вращения, образованных различными кривыми. Общий вид уравнения кривой выглядит как

$$y = a \cdot x^2 + 2 \cdot (b+c) \cdot x + d \cdot \ln(x+4).$$

Зная количество тел вращения, уравнения кривых и границы отрезков, на которых эти кривые применяются, найдите массу сердечника. Сердечник сделан из обеднённого урана.

Формат ввода

В строке вводится натуральное число N – число тел вращения. N не превышает 5.

Далее следует N шестёрок вещественных неотрицательных чисел x_1, x_2, a, b, c, d – соответственно, координаты границ отрезка, и коэффициенты при кривой на этом отрезке. Все числа отделены друг от друга одним или несколькими пробелами.

Гарантируется, что разрывов нет и каждый следующий отрезок начинается там, где закончился предыдущий.

Гарантируется, что каждое уравнение имеет математический смысл.

Никакие числа не превышают **1000**.

Формат вывода

На выходе программа должна выдать целое число – массу сердечника в граммах, округлённую до ближайшего целого.

Плотность обеднённого урана – $19,05 \text{ г/см}^3$. Все значения переменных даны в сантиметрах.

Примечание: число требуется вывести с точностью до целого только ради исключения накладок при тестировании. Рекомендуется считать суммы с заранее известной точностью $\varepsilon = 10^{-5}$. Точность ε считать достигнутой, когда при вычислении интегральной суммы уменьшение отрезка x вдвое приводит к изменению суммы меньше, чем на ε

Пример

Входные данные	Выходные данные
1 0 1 1 1 1 1	1017

Решение

```
program z17;
const
  ro=19.05;
  pi=3.1415926;
  eps = 0.00001;

function f(x,a,b,c,d:real):real;
begin
  f:=a*x*x+2*(b+c)*x+d*ln(x+4);
end;

function work(x0,x1,a,b,c,d:real):real;
var vt,pv,x,h:real;
begin
  vt:=sqr(f((x1+x0)/2,a,b,c,d))*(x1-x0);
  h:=(x1-x0)/2;
  repeat
    pv:=vt;
    x:=x0;
    vt:=0;
    while x<x1 do
      begin
        vt:=vt+sqr(f(x,a,b,c,d))*h;
        x:=x+h;
      end;
    h:=h/2;
  until abs(vt-pv)<eps;
  work:=ro*pi*vt;
end;

var w,x0,x1,a,b,c,d:real;
```

```

    i,n:integer;
begin
w:=0;
read(n);
for i:=1 to n do
begin
read(x0);
read(x1);
read(a);
read(b);
read(c);
read(d);
w:=w+work(x0,x1,a,b,c,d);
end;
writeln(round(w));
end.

```

Тесты

N	Оценка	Входные данные	Выходные данные
1	10	1 0 1 1 0 0 0	12
2	10	2 0 1 1 1 1 1 1 2 1 2 1 1	11609
3	10	3 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1	3050