

Второй (заключительный) этап академического соревнования

Олимпиады школьников «Шаг в будущее»

по общеобразовательному предмету «Информатика»

10 класс, февраль, 2016 г.

Вариант № 2.

Задание 1 (12 баллов)

Определить основание системы счисления, в которой записано выражение:

$$\begin{array}{r} aba_y \\ + b4_y \\ \hline b00_y \end{array}$$

где a и b - цифры числа.

Ответ: 5

Решение. Исходя из формулы, запишем систему уравнений.

$$a+4=y$$

$$b+b+1=y$$

$$a+1=b$$

Очевидно, что $a+4=2a+3$, следовательно, $a=1$. $y=1+4=5$.

Критерии оценки.

На полный балл (12 баллов) оценивается решение, в котором в явном виде приведено уравнение или цепочка рассуждений, ведущая к обоснованию правильного ответа.

Решение, которое содержит арифметические ошибки, не ведущие к нарушению цепочки рассуждений, оценивается в 8 баллов.

Решение, содержащее только правильный ответ, оценивается в 3 балла.

Задание 2 (12 баллов)

- Мои четыре внучки - замечательные девочки, - рассказывала бабушка Пелагея с нескрываемой гордостью. Каждая из них играет на каком-нибудь музыкальном инструменте и говорит на одном из иностранных языков.

- На чем играет Маша? - спросил я.

- На рояле.

- А кто играет на скрипке?

- Помню только что это та девочка, которая говорит по-французски - ответила бабушка.

Поговорив с бабушкой, я также узнал, что Оля играет на виолончели, а Лена не говорит по-немецки. Маша не знает итальянского языка, а Оля не владеет английским. Валя не знает французского, Лена не играет на арфе, а виолончелистка не говорит по-итальянски. Я совсем запутался. Скажите, кто на каком инструменте играет, и на каком языке говорит?

Критерии:

- -5 баллов за каждое несоответствие

Решение:

1. Так как на арфе играет не Маша, не Оля, и не Лена, значит на арфе играет Валя, а Лена играет на скрипке.
2. Т.к. Лена играет на скрипке, то она говорит по-французски.
3. Оля не говорит по-английски. Она играет на виолончели, следовательно не говорит по-итальянски. Она также не говорит по-французски, т.к. по-французски говорит Лена. Значит Оля говорит по-немецки.
4. Т.к. Маша не говорит по-итальянски, по-французски и по-немецки, она говорит по-английски, а Валя по-немецки.

| | Рояль | Скрипка | Арфа | Виолончель | Фр. | Нем. | Ит. | Англ. |
|------|-------|---------|------|------------|-----|------|-----|-------|
| Маша | + | - | - | - | - | - | - | + |
| Оля | - | - | - | + | - | + | - | - |
| Лена | - | + | - | - | + | - | - | - |
| Валя | - | - | + | - | - | - | + | - |

Задание 3 (12 баллов)

Дед Мороз раздал подарки на Новый год 30 детям. 15 ребят рассказали стишок. 10 ребят спели песенку. 7 ребят спели песенку и сплясали. 5 ребят рассказали стишок и сплясали. 3 ребёнка пели, и плясали, и декламировали стишок. Сколько ребят сплясало, если известно, что просто так подарок не дали никому?

Ответ: 14

Решение.

Всего пели песенку 10 человек. Известно, что 7 при этом сплясали, 3 и пели, и плясали, и рассказывали стишок. Отсюда следует, что детей, которые бы пели песенку и при этом не плясали, нет. Значит, подставим по формуле Байеса числа.

$$X+15+10-7-5+3=30. X+16=30, X=14.$$

Критерии. На полный балл оценивается решение, в котором в явном виде приведены круги Эйлера или цепочка рассуждений, ведущая к обоснованию правильного ответа.

Решение, которое содержит арифметические ошибки, не ведущие к нарушению цепочки рассуждений, оценивается в 8 баллов

Решение, содержащее только правильный ответ, оценивается в 5 баллов во всех классах.

Задание 4 (12 баллов)

На вход автомата подаётся четырёхзначное число в восьмеричной системе счисления. Автомат выполняет следующие действия:

- переводит число в двоичную систему счисления;
- все разряды числа (двоичного) переписывает в обратном порядке;
- переводит число в шестнадцатеричную систему счисления.

Укажите максимальное число (в восьмеричной системе счисления), такое, что подав его на вход автомата, мы получим трёхзначное число (в восьмеричной системе) у которого все цифры чётные.

Решение

т.к. надо получить 3-хзначное число то младшая цифра исходного числа должны быть равна 0 остальные должны дать 0 в младшем разряде триады после поворота т.е. 3330

Критерии (макс. 12 баллов)

- -6 если ответ не в той системе счисления
- -6 если ответ 3333
- -8 если ответ 2220
- -10 если ответ 1110
- -4 нет решения

Задание 5 (12 баллов)

В кабине лифта 20-этажного дома есть 2 кнопки. При нажатии на одну из них лифт поднимается на 13 этажей, при нажатии на другую — опускается на 8 этажей. Как попасть с 13-го этажа на 8-й?

Решение

| № пп | Начальное состояние (номер этажа) | Нажатие кнопки | Конечное состояние (номер этажа) |
|---------|---|-------------------|--|
| 1 | 13 | -8 | 5 |
| 2 | 5 | +13 | 18 |
| 3 | 18 | -8 | 10 |
| 4 | 10 | -8 | 2 |
| 5 | 2 | +13 | 15 |
| 6 | 15 | -8 | 7 |
| 7 | 7 | +13 | 20 |
| 8 | 20 | -8 | 12 |
| 9 | 12 | -8 | 4 |
| 10 | 4 | +13 | 17 |
| 11 | 17 | -8 | 9 |
| 12 | 9 | -8 | 1 |

| | | | |
|----|----|-----|----|
| 13 | 1 | +13 | 14 |
| 14 | 14 | -8 | 6 |
| 15 | 6 | +13 | 19 |
| 16 | 19 | -8 | 11 |
| 17 | 11 | -8 | 3 |
| 18 | 3 | +13 | 16 |
| 19 | 16 | -8 | 8 |

Критерии

Оценивается только решение, приведшее к правильному результату.

Если количество нажатий на кнопку превышает количество нажатий на кнопку эталонного решения, то вычитается 3 балла за каждое лишнее нажатие на кнопку.

Задание 6 (15 баллов)

Классный руководитель хочет посмотреть, насколько соотносятся оценки выпускников школы с результатами сдачи ЕГЭ по информатике. Для этого ему нужно знать средний балл двоечников, троечников, хорошистов и отличников. Однако электронный журнал находится на профилактике, и поэтому он получает данные через соцсеть. Требуется найти средний балл каждой категории учащихся.

Формат входных данных:

На первой строке задаётся целое число N - количество ответов на запрос классного руководителя

В следующих N строках записан ответ учащегося. Ответ гарантированно содержит ровно два целых неотрицательных числа, ограниченных пробелом или концом строки. Первое число является оценкой школьника по предмету, второе - количеством баллов по ЕГЭ.

Вам необходимо определить средние баллы каждой из четырёх категорий учащихся. Если учащихся такой категории среди ответов нет, средний балл равен 0.

Пример:

Входные данные:

4

Оценка 5, баллов 86.

Оценка 4, баллов 75

4 и 73

Оценка 3, а вот баллов 76

Выходные данные:

0 76 74 86

Критерии оценки. В 15 баллов оценивается верно работающее решение задачи, удовлетворяющее следующим критериям:

- время работы программы пропорционально количеству входных данных;
- входные данные не хранятся в массиве целиком

Не оптимальное по расходу памяти решение (например, записать все в массив и найти 4 максимума) оценивается в 11 баллов.

Не оптимальное по времени работы решение (например, записать все в массив и отсортировать его) оценивается в 7 баллов.

Неверно работающая программа, из которой, тем не менее, понятно, что участник понимает решение задачи, оценивается в 3 балла.

//по очереди читаем строчки

//извлекаем из каждой строки оценку и балл, после чего

//считаем средние арифметические

```
program league;
```

```
const M=5;
```

```
type tmas=array [2..M] of integer; //массив, индексы которого совпадают с оценками
```

```
procedure cut(s:string; var x,y:integer);
```

```
var s1:string;
```

```
    flag1,flag2:boolean;
```

```
    code:integer;
```

```
begin
```

```
s:=s+' ';
```

```
while pos(' ',s)>0 do
```

```
    delete(s,pos(' ',s),1); //чистим лишние пробелы
```

```
    flag1:=true; //истина, пока не взято 1 число
```

```
    flag2:=true; //истина, пока не взято 2 число
```

```
while flag1 do //пока не взято первое число
```

```
    begin
```

```
        s1:=copy(s,1,pos(' ',s)-1); //извлекаем слово
```

```
        delete(s,1,pos(' ',s));
```

```
        val(s1,x,code); //пытаемся преобразовать слово в оценку
```

```
        if code=0 then //если результат удачный, то опускаем флаг
```

```
            begin
```

```
                flag1:=false;
```

```
            end;
```

```
end;
```

```
while flag2 do
```

```

begin
  s1:=copy(s,1,pos(' ',s)-1);
  delete(s,1,pos(' ',s));
  val(s1,y,code); //пытаемся преобразовать слово в средний балл
  if code=0 then //если результат удачный, то опускаем флаг
    begin
      flag2:=false;
    end;
  end;
end;

```

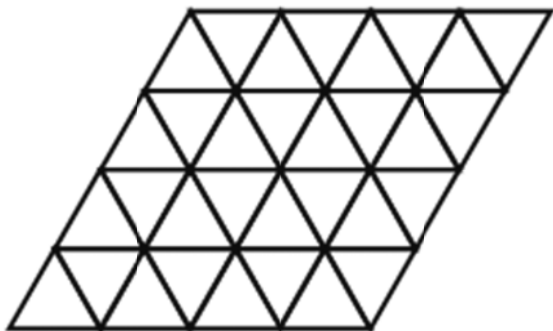
```

var n,x,y,i:integer;
    ball,kolvo:tmas;
    s,name:string;
begin
  readln(n); //читаем количество строк
  for i:=2 to M do //инициализация переменных
    begin
      ball[i]:=0;
      kolvo[i]:=0;
    end;
  for i:=1 to n do
    begin
      readln(S); //читаем очередную строчку
      cut(S,x,y); //извлекаем два числа - оценку и баллы по ЕГЭ
      ball[x]:=ball[x]+y; //добавляем их в массивы
      kolvo[x]:=kolvo[x]+1;
    end;
  for i:=2 to M do
    begin
      if kolvo[i]<>0 then
        write(ball[i]/kolvo[i],' ') //выводим среднее
      else
        write(0,' ');
      end;
    end.

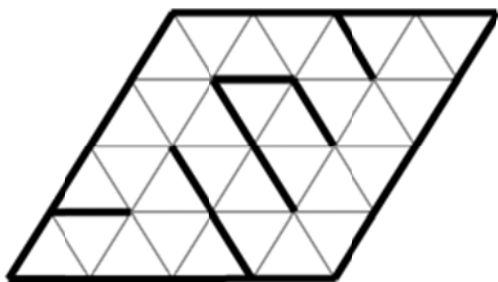
```

Задание 7 (25 баллов)

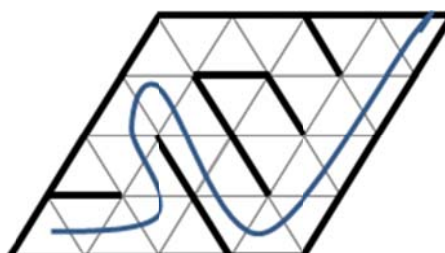
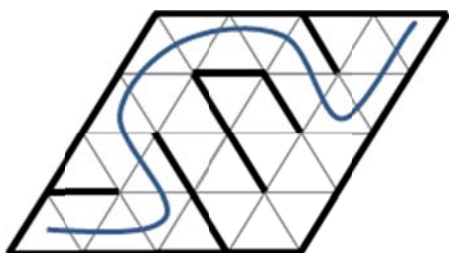
Некоторый замок состоит только из комнат, которые представляют собой равносторонние треугольники (см рис.1)



Не в каждой из стен комнат есть двери. Обозначим на рисунке стены, в которых есть двери тонкими линиями, а в которых нет дверей толстыми.



Посетителю замка необходимо добраться из левого нижнего угла в правый верхний, посетив наименьшее количество комнат.



Выше приведены рисунки для двух способов перемещения в конечную комнату. В первом случае путь включает в себя 18 комнат, а во втором 20 (начальная и конечная комната учитываются).

Необходима написать программу, которая для заданного замка находит кратчайший путь и выводит количество комнат, которые необходимо посетить посетителю замка.

Исходные данные берутся из файла. В первой строке файла через пробел записываются количество рядов комнат N и количество комнат в ряду M .

В следующих $2*N+1$ строках описываются стены комнат – стена с дверью обозначается цифрой 0, без двери – цифрой 1. Для замка, приведённого на рисунке, файл будет иметь следующий вид.

4 8

1111

100001001

0100

100101001

0000

100101001

1000

100100001

1111

Ответ

18

Правильно работающее решение задачи оценивается в 25 баллов.

Если программа работает, но находит не оптимальный, а некий другой путь, который не нарушает остальных правил, то такая программа оценивается из 20 баллов.

За ввод с клавиатуры (вместо файла) вычитается 2 балла. Синтаксические ошибки, которые не влияют на логику работы программы не учитываются (например, begun, than, пропущенные точка с запятой и т.п.). Если программа работает в целом правильно, но есть ошибки типа or вместо and, больше вместо меньше и наоборот и т.п., то такие ошибки наказываются вычитанием по 2 балла за каждую, но не более 3-х ошибок такого рода. Если таких ошибок больше, то считается, что в целом такая работа работает неправильно.

Если программа в целом работает не правильно, то оцениваем отдельные части программы, а именно

1. Если программа считывает данные и создаёт адекватную структуру, например, массив, в котором фиксируются все необходимые данные (стены замка, наличие или отсутствие дверей), то данная часть программы оценивается в 5 баллов
2. Если программа правильно выполняет волну (или другой оптимальный поиск), то эта часть программы оценивается в 10 баллов
3. Если программа может строить правильный, но не оптимальный путь, то эта часть программы оценивается в 5 баллов.

{Идея решения

Каждая комната имеет три стены. Будем для каждой стены записывать

координаты связанной с ней комнаты, если в стене есть дверь
и записывать нули, если двери нет (то есть через данную стену комната не связана с другими)
Затем запускаем классическую волну, учитывая, что соседними комнатами для текущей
комнаты будут те, с которыми указана связь (то есть не нули)

}

const

row = 10;

col = 20;

type

coord = **record**

rw, cl: **integer**;

end;

link = **array**[1..3] **of** coord;

pole = **record**

inf: **integer**;

p: link;

end;

mas = **array**[1..row, 1..col] **of** pole;

procedure init(**var** zamok: mas);

var

i, j, k: **integer**;

begin

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

for j := 1 **to** col **do**

with zamok[i, j] **do**

begin

inf := 0;

for k := 1 **to** 3 **do**

begin

p[k].rw := 0; p[k].cl := 0;

end

end;

end;

procedure razb0(**var** zamok: mas; k: **integer**; s: **string**);

```

var
  i:integer;
begin
  for i := 1 to length(s) do
    if copy(s, i, 1) = '0' then //если в текущей стене дверь, то
      begin
        zamok[k div 2, 2 * i - 1].p[1].rw := k div 2 + 1;
        zamok[k div 2, 2 * i - 1].p[1].cl := 2 * i;
        zamok[k div 2 + 1, 2 * i].p[1].rw := k div 2;
        zamok[k div 2 + 1, 2 * i].p[1].cl := 2 * i - 1;
      end;
    end;
  procedure razb1(var zamok: mas; k: integer; s: string);
  var
    i:integer;
  begin
    for i := 1 to length(s) do
      if copy(s, i, 1) = '0' then //если в текущей стене дверь, то
        begin
          zamok[k div 2 + 1, i-1].p[2].rw := k div 2 + 1;
          zamok[k div 2 + 1, i-1].p[2].cl := i;
          zamok[k div 2 + 1, i ].p[3].rw := k div 2 + 1;
          zamok[k div 2 + 1, i ].p[3].cl := i-1;
        end;
      end;
    procedure inp(var f: text; var zamok: mas; n: integer);
  var
    i: integer;
    s: string;
  begin
    readln(f); //первую строчку пропускаем (внешняя стена замка - дверей нет)
    for i := 1 to 2 * n - 1 do //перебираем все остальные строчки, кроме последней
      begin
        readln(f, s); //считываем строку с дверьми
        if i mod 2 = 0 then //если строка четная, то
          razb0(zamok, i, s) //вносим наличие дверей в горизонтальных (по рисунку) стенках
        else
          razb1(zamok, i, s); //вносим наличие дверей в вертикальных (по рисунку) стенках
      end;
    end;
  end;
end;

```

```

end;
end;
procedure volna(zamok:mas; n, m:integer;var res:integer);// запускаем волну
var
i,j,k:integer;
begin
zamok[n,1].inf:=1; //ставим 1 в начало пути
res:=1;
repeat //повторяем
for i:=1 to n do //перебираем все комнаты
for j:=1 to m do
with zamok[i,j] do
begin
if inf=res then //если в текущую комнату попали на предыдущем ходе ,то
for k:=1 to 3 do //перебираем стены
begin
if p[k].rw<>0 then //если есть дверь, то
if zamok[p[k].rw,p[k].cl].inf=0 then //если еще не были в соседней комнате
zamok[p[k].rw,p[k].cl].inf:=res+1; //делаем шаг в соседнюю комнату
end;
end;
end;
inc(res);
until zamok[1,m].inf<>0 //пока не дойдем до конечной точки
end;
var
f: Text;
n, m, res: integer;
zamok: mas;
begin
Assign(f, 'input.txt');
Reset(f);
init(zamok); //обнуляем
readln(f, n, m);// считываем размеры
inp(f, zamok, n);// считываем конфигурацию
close(f);
volna(zamok, n, m, res);// запускаем волну
writeln(res);// выводим результат
end.

```