

**Межрегиональная предметная олимпиада КФУ
по предмету «Информатика»
заключительный этап (решения)
2018-2019 учебный год
9 класс**

№1. Кубик. (25 баллов)

На вертикальной клетчатой доске размером N (высота) на M (ширина) клеток находится магнитная игральная кость (кубик). Она расположена в левой верхней клетке (с координатами 1, 1) так, что ее грань на доске занимает ровно одну клетку, т.е. каждая грань кости равна по размерам одной клетке. Напротив грани кости с числом 1 написано число 6, напротив числа 2 – грань с числом 5 и напротив числа 3 написано 4. За один ход кость перекачивается через ребро грани на доске (перекапывается, а не сдвигается) в соседнюю клетку, не выходя за рамки доски, и занимает всю соседнюю клетку. Теперь одна из боковых граней находится на доске. При этом ведется подсчет очков тех граней, которые соприкасаются с клеткой, т.е. на которую примагничена кость в данный момент. Вычислить и напечатать минимальную сумму очков, которая наберется, если прокатить игральную кость в правый нижний угол листа (с координатами N, M). В начальной позиции верхняя грань кости с числом 1, левая грань с числом 2, а грань на доске – 3. Исходные данные – натуральные числа, перечисленные через пробел, N и M – размеры листа ($1 < N, M < 100$).

Входные данные:	Выходные данные
2 2	14

Решение.

Похожая, но более общая, задача имеется на сайте asm.timus.ru (номер 1016 «кубик на прогулке»). В ней на каждой грани может быть записано произвольное натуральное число.

Для ее решения представим матрицу как неориентированный граф с весами ребер, где вершина – это каждая клетка с учетом текущего положения кубика (т.е. с учетом поворота граней), а ребро соединяет смежные вершины (клетки) и имеет вес, равный числу на новой нижней грани кубика. Далее, можно использовать один из известных алгоритмов поиска кратчайшего пути из начальной вершины в конечную – алгоритм Дейкстры, Форда-Беллмана, поиск в ширину, поиск в глубину и другие. Так как начальное положение граней кубика не известно (какая грань внизу, какие по бокам), следует выполнить этот процесс для каждого варианта расположения кубика.

В нашем, более простом случае, можно заметить, что положение кубика повторяется после нескольких (сколько точно?) поворотов по горизонтали и по вертикали. Остается рассмотреть небольшой набор значений для n и m .

Баллы

- 25 баллов за полное и оптимальное решение.
- 15 баллов за неполное решение.
- 8 баллов за наличие идей решения.

№2. Выборы. (10 баллов)

В выборах Президента Страны участвуют N кандидатов. Проведенный социологический опрос показал, что за i -го кандидата хотят проголосовать A_i избирателей. Кандидат номер 1 хочет победить наверняка и решается подкупить некоторое число избирателей, чтобы они проголосовали за него. Для подкупа одного любого избирателя нужно дать ему одну конфетку. Какое минимальное число конфеток должен приготовить Кандидат номер 1, чтобы набрать голосов больше, чем любой другой кандидат. $1 \leq N \leq 1000, A_i < 1000$.

В первой строке задано N и во второй строке N целых чисел, перечисленных через пробел, – количество избирателей голосующих за кандидатов.

Входные данные:	Выходные данные
5 2 11 7 14 13	9

Решение.

```

Var i,n,j,k,a1,b:integer;
  A:array[2..1000] of integer;
Begin
  Read(n,b); a1:=b;
  For i:=2 to n do read(A[i]);
  For i:=n downto 3 do
    For j:=2 to i-1 do
      Begin k:=a[j];a[j]:=a[j-1]; a[j-1]:=k end;
  K:=2;r:=a[2];
  While a[1]<=A[k] do begin
    A[k]-=1;
    a1+=1;
    k+=1;
    if (a[k]<r) or (k>n) then k:=2
  End;
  Write(a1-b)
End.

```

Баллы

- 10 баллов за полное и оптимальное решение.
- 7 баллов за неполное решение.
- 3 баллов за наличие идей решения.

№3. Контрразведка. (15 баллов)

Четыре разведывательные группы хранили общую секретную информацию в одной камере хранения. Каждая группа записывала шифр в своей системе исчисления. Первая группа в двоичной системе, 2-ая группа – 4-чной системе, 3-я группа – 8-чной системе, 4-я группа – 16-чной системе. Контрразведчики, наблюдая за разведчиками, узнали некоторые цифры шифра.

Определить минимальное число вариантов перебора кодов ключа, позволяющих вскрыть сейф с информацией.

В четырех строчках приведены обнаруженные коды контрразведчиков. Длина кодов не превышает 200 символов. Не известные разряды кода задаются символом ”*”.

Входные данные:	Выходные данные
*0***** *****2 **0* *8*	16

Решение.

```

Var c,c2,c4,c8,c16:string;
  i,k,n,d:integer;
  r4:array[‘0’..‘3’] of string=(‘00’,‘01’,‘10’,‘11’);
  r8:array[‘0’..‘7’] of string=(‘000’,‘001’,‘010’,‘011’,‘100’,‘101’,‘110’,‘111’);

```

```
r16:array['0'..'9','A'..'F'] of string=
      ('0000','0001','0010','0011','0100','0101','0110','0111',
       '1000','1001','1010','1011','1100','1101','1110','1111');
```

```
Begin
  Readln(c2);
  Readln(c4);
  Readln(c8);
  Readln(c16);
  N:=max(max(length(c2),2* length(c4)), max(3*length(c8),4*length(c16)));
  C2[1]:= '1'; // старший значащий разряд – '1'
  d:=length(c2); // число 2-ичных значащих разрядов
  For i:= d+1 to n do c2:='*' +c2;
  // разбор c4
  For i:=length(c4) downto 1 do
    If c4[i]<>'*' then begin
      C2[n-2*(i-1)]:=r[c4[i]][1];
      C2[n-2*(i-1)-1]:=r[c4[i]][2]
    End;
  // аналогично разбор c8
  // аналогично разбор c16
  S:=1;
  For i :=length(c2) downto n-d+1 do
    If c2[i]='*' then s*=2;
  Write(s)
End.
```

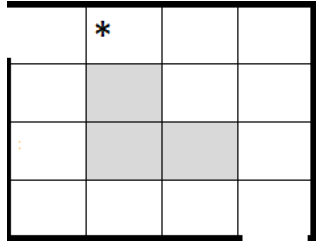
Баллы

- 15 баллов за полное и оптимальное решение.
- 10 баллов за неполное решение.
- 5 баллов за наличие идей решения.

№4. Лабиринт. (20 баллов)

Дан лабиринт размерности $N \times N$, по которому можно ходить направо или вниз. Из клетки со звездочкой "*" можно делать телепортацию в любую клетку по диагонали к выходу, где нет препятствий при перемещении (с клетки A,B в клетку A+K,B+K). Написать программу, которая по заданному расположению стенок и проходов в клетках определит число путей, ведущих из левого верхнего входа к нижнему правому выходу.

Входные данные. В первой строке задается N ($N < 100$). Далее в N строках заданы строки лабиринта, где клетка телепортации задается «*», проходная клетка «.», а не проходная «X».

Входные данные:	Выходные данные	
4	5	
.*..		
.X..		
.XX.		
....		

Решение.

```
Var i,j,k,:integer;
  L:array[1..100,1..100] of char;
  P:array[0..100,0..100] of integer;
```

```

Begin
Read(n);
For i:=0 to 100 do begin P[i,0]:=0; P[0,i] end;
For i:= to n do begin
  For j:=1 to n do begin read(L[i,j]); P[i,j]:=0 end;
  Readln
End;
For i:=1 to n do
  For j:= to n do
    If i+j=2 then P[I,j]:=1 else
    If L[i,j]<>'X' then begin
      P[I,j] +=P[i-1,j]+P[I,j-1];
      K:=2;
      While (k<min(i,j)) and (L[i-k,j-k]<>'X') do begin
        If L[i-k,j-k]='*' then P[i,j]+=P[i-k,j-k];
        K+=1
      end
    End;
  Write(P[n,n])
End.

```

Баллы

- 20 баллов за полное и оптимальное решение.
- 13 баллов за неполное решение.
- 7 баллов за наличие идей решения.

№5. Слоги. (15 баллов)

Задан текст на русском языке.

Написать программу вычисления средней длины слога в этом тексте с точностью до сотых долей.

В русском языке 13 типов слогов, где слога одного из типов являются словами из одной согласной буквы, а остальные типы слогов содержат одну согласную букву.

PS. слова состоят из слогов, а слога из букв.

Входные данные:	Выходные данные
Попрыгунья стрекоза лето красное пропела.	2.40

Решение.

```

#import <bits/stdc++.h>
using namespace std;
main(){
  int i,j,dl=0,k=0;
  char s[20];
  while (cin >> s) {
    int c=0, d;
    d=strlen(s)-1+alpha(s[strlen(s)-1]);
    dl+=d;
    k+=d=1;
    if(d>1)
      for(i=0;i<d;i++)
        if (s[i]=='a' | s[i]=='A' | ...) k++; // проверка на гласность
  }
}

```

```
cout << setprecision(6) << dl/k;
}
```

Баллы

- 15 баллов за полное и оптимальное решение.
- 10 баллов за неполное решение.
- 5 баллов за наличие идей решения.

№6. Раз, два... (15 баллов)

Сколько N-значных чисел можно составить, используя цифры 1 и 2, в которых три одинаковые цифры не стоят рядом? Вход – число N ($1 \leq N \leq 30$).

Выход – нужно вывести одно число – количество чисел с указанным свойством.

Входные данные:	Выходные данные
4	10

Решение.

Простой анализ для небольших значений N показывает, что ответ равен удвоенному очередному числу Фибоначчи. Например, для $N = 1$ ответ 2, для $N = 2$ ответ 4, для $N = 3$, ответ 6. Последовательность чисел Фибоначчи 1, 2, 3, 5, 8, 13, 21, ... строится по простому принципу: первые два числа задаются, а каждое следующее вычисляется (оно равно сумме двух предыдущих).

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    cin >> n;
    int f1=0, f2=1;
    for(int i=0; i<n; i++)
    {
        swap(f1, f2);
        f2 += f1;
    }
    cout << f2 * 2;
    return 0;
}
```

2-ой вариант. Перебор всех N-разрядных чисел с цифрами 1, 2. Далее подсчет количества чисел, не содержащих 3 подряд идущих единиц или двоек.

```
Var s:string;
    l,k:integer;
Begin
    Read(n);
    sC:='';
    For i:=1 to n do s:=s+'1'
    k:=0;
    Repeat
        i:=n;
        while (s[i]='2') and (i>0) do begin
            s[i]:=1;
```

```
i:=1
end;
if i>0 then s[i]:= '2';
k+=(pos('111',s)=0) and (pos('222',s)=0)
Until i=0;
Write(k)
End.
```

Баллы

- 15 баллов за полное и оптимальное решение.
- 10 баллов за неполное решение.
- 5 баллов за наличие идей решения.