

Разбор задач олимпиады «Муниципальный этап Всероссийской олимпиады школьников Красноярского края по информатике, 9-11 классы» сайта астр.ru от 07.12.2013 года

Задача А. Дорожная политика

Количество различных дорог равняется количеству способов выбрать два города из N . В комбинаторике это обозначается C_N^2 и находится по формуле:

$$C_N^2 = \frac{N!}{(N-2)! \cdot 2!} = \frac{N \cdot (N-1)}{2}.$$

Все недополученный баллы лежат в переполнении типов. Для данной задачи не хватает даже типов `long long` и `Int64`. Поэтому в C++ следует воспользоваться `unsigned long long`. А в Паскале сначала поделить, а только потом умножить.

Решение на языке C++

```
1 #include <iostream>
2 #include <cstdio>
3
4 using namespace std;
5
6 int main(){
7     freopen("input.txt", "r", stdin);
8     freopen("output.txt", "w", stdout);
9
10    unsigned long long n;
11    cin>>n;
12    cout<<n*(n-1)/2;
13
14    return 0;
15 }
```

Решение на языке Паскаль

```
1 var
2   n : Int64;
3 begin
4   assign(input, 'input.txt'); reset(input);
5   assign(output, 'output.txt'); rewrite(output);
6
7   Read(n);
8   if n mod 2 = 0 then Write(n div 2 * (n-1))
9   else Write((n-1) div 2 * n);
10 end.
```

Задача В. Спичрайтер Йоды

Будем считывать по одному слову и складывать их в стек. Как только в конце очередного слова замечаем точку, следует вывести все слова из стека (в обратном порядке, именно поэтому и используется стек). А считывание продолжить дальше.

Особым моментом является вывод пробелов между словами и предложениями (а также в

начале и в конце строки). При тестировании задачи необходимо обратить на это внимание и, возможно, добавить условие (в решении на С++ это строки 18 и 19).

При решении на языке Паскаль дополнительной сложностью является отсутствие встроенных возможностей считывания по словам, а также ручная реализация стека. Но количество строк в итоге, всё равно, получается меньше.

Решение на языке С++

```
1 #include <iostream>
2 #include <cstdio>
3 #include <cstring>
4 #include <stack>
5
6 using namespace std;
7
8 int main(){
9     freopen("input.txt", "r", stdin);
10    freopen("output.txt", "w", stdout);
11
12    string str;
13    stack<string> st;
14    bool f=false;
15    while(cin>>str){
16        if(str[str.size()-1]==''){
17            str=str.substr(0, str.size()-1);
18            if(f) cout<<" ";
19            else f=true;
20            cout<<str;
21            while(!st.empty()){
22                cout<<" "<<st.top();
23                st.pop();
24            }
25            cout<<".";
26        }else st.push(str);
27    }
28
29    return 0;
30 }
```

Решение на языке Паскаль

```
1 var
2   s : AnsiString;
3   ms : array [0..20000] of AnsiString;
4   i, lm, j : Longint;
5   f : Boolean;
6 begin
7   assign(input, 'input.txt'); reset(input);
8   assign(output, 'output.txt'); rewrite(output);
9
10  ReadLn(s);
11  lm := 0;
12  f := false;
13  ms[0] := '';
14  for i:=1 to length(s) do
15    if s[i] = '.' then begin
16      if f then Write(' ')
17      else f := true;
18      Write(ms[lm]);
19      for j:=lm-1 downto 0 do Write(' ', ms[j]);
20      Write('.');
21      lm := -1;
22    end else begin
23      if s[i] <> ' ' then ms[lm] := ms[lm] + s[i]
24      else begin
25        inc(lm);
26        ms[lm] := ' ';
27      end;
28    end;
```

Задача С. Прыжки в длину

Считывание данных в этой задаче для «паскалистов» выглядит очень непросто. Тут надо уметь не только выделять слова в строке, но и исключать лишние пробелы. А также преобразовать символьное представление числа в числовой тип данных. Заступ удобно заменить на число -1 .

Для нахождения ТОП-3 при маленьких ограничениях удобно отсортировать участников. Но заметим, что поиск ТОП-3 можно осуществить и за $O(N)$.

Так при сортировке участников нам придётся сравнивать шесть их попыток в порядке убывания, то для каждого участника тоже предварительно можно отсортировать его попытки. После этого для определения лучшего среди двух надо будет сначала сравнить шестые попытки, если они равны, то – пятые, и так далее.

В решении на С++ опять же используется стандартная сортировка с компаратором, а в решении на Паскале в обоих случаях применяется Гномья сортировка.

Обратите внимание, что в решении на Паскале сами участники не сортируются, а переставляются лишь их порядковые номера в отдельном массиве. Это немного усложняет обращение к данным, но экономит время на копировании всей информации об участнике из одной строки массива в другую.

Решение на языке С++

```
1 #include <iostream>
2 #include <cstdio>
3 #include <algorithm>
4
5 using namespace std;
6
7 int mm[109][6], m[109];
8
9 bool cmp(int a, int b){
10     int i;
11     for(i=5;i>=0;--i) if(mm[a][i]!=mm[b][i]) break;
12     return ((i>=0 && mm[a][i] > mm[b][i]));
13 }
14
15 int main(){
16     freopen("input.txt", "r", stdin);
17     freopen("output.txt", "w", stdout);
18
19     int n;
20     char ss[109][3][99];
21
22     cin>>n;
23     for(int i=0;i<n;++i){
24         for(int j=0;j<3;++j) cin>>ss[i][j];
25         for(int j=0;j<6;++j){
26             char c;
27             cin>>c;
28             if(c=='x') mm[i][j]=-1;
29             else{
30                 int a;
31                 mm[i][j] = (c-'0')*100;
32                 cin>>c>>a;
33                 mm[i][j]+=a;
34             }
35         }
36         sort(mm[i], mm[i]+6);
37         m[i]=i;
38     }
39     sort(m, m+n, cmp);
40
41     bool f=false;
```

```
42     for(int i=0;i<3 && i<n;++i){
43         if(mm[m[i]][5]>=0){
44             cout<<i+1<<" " <<ss[m[i]][0]<<" " <<ss[m[i]][1]<<" " <<ss[m[i]][2]<<" " <<mm[m[i]][5]/100<<". ";
45             if(mm[m[i]][5]%100<10)cout<<0;
46             cout<<mm[m[i]][5]%100<<endl;
47             f=true;
48         }
49     }
50
51     if(!f) cout<<"No results.\n";
52     return 0;
53 }
```

Решение на языке Паскаль

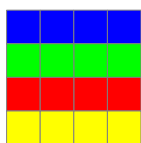
```
1  uses math;
2  var
3      n, i, j, c, t : Longint;
4      ss : array [0..109] of AnsiString;
5      s, s1 : AnsiString;
6      mm : array [0..109,0..5] of Double;
7      m : array [0..109] of Longint;
8      f : Boolean;
9      d : Double;
10 function check(a, b : Longint) : Boolean;
11 var
12     i : Longint;
13 begin
14     i := 5;
15     while i>=0 do begin
16         if mm[a, i] <> mm[b, i] then break;
17         dec(i);
18     end;
19     check := (i>=0) and (mm[a, i] < mm[b, i]);
20 end;
21
22 begin
23     assign(input, 'input.txt'); reset(input);
24     assign(output, 'output.txt'); rewrite(output);
25
26     ReadLn(n);
27     for i:=0 to n-1 do begin
28         ReadLn(s);
29         s := s + ' ';
30         c := 0;
31         j := 1;
32         while c<3 do begin
33             ss[i] := ss[i] + s[j];
34             inc(j);
35             if (s[j] = ' ') and (s[j-1] <> ' ') then inc(c);
36         end;
37         for c:=0 to 5 do begin
38             while s[j] = ' ' do inc(j);
39             s1 := '';
40             while s[j] <> ' ' do begin
41                 s1 := s1 + s[j];
42                 inc(j);
43             end;
44             if s1='x' then mm[i, c] := -1
45             else val(s1, mm[i, c], t);
46         end;
47         m[i] := i;
48         j := 0;
49         while j<5 do
50             if mm[i, j] > mm[i, j+1] then begin
51                 d := mm[i, j];
52                 mm[i, j] := mm[i, j+1];
53                 mm[i, j+1] := d;
54                 if j>0 then dec(j);
55             end else inc(j);
56         end;
57     end;
58     i:=0;
```

```
59   while i<n-1 do
60       if check(m[i], m[i+1]) then begin
61           t := m[i];
62           m[i] := m[i+1];
63           m[i+1] := t;
64           if i>0 then dec(i);
65       end else inc(i);
66
67   f := false;
68   for i:=0 to min(3, n)-1 do
69       if mm[m[i]][5] >= 0 then begin
70           WriteLn(i+1, ' ', ss[m[i]], ' ', mm[m[i], 5]:0:2);
71           f := true;
72       end;
73
74   if not f then WriteLn('No results.');
```

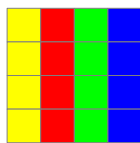
Задача D. Офисы

Это единственная идейная задача олимпиады. Только в этой задаче надо было что-то придумать, а не реализовать какой-либо стандартный алгоритм или формулу. Рассмотрим один из вариантов решения.

Можно заметить, что для размещения фирм достаточно всего двух этажей, если их поселить следующим образом (офисы одной фирмы покрашены в один цвет):



этаж 1



этаж 2

Все остальные этажи можно заполнить любым цветом, либо можно просто продублировать второй этаж до самого верха здания.

После заполнения осталось лишь в правильном порядке вывести трёхмерный массив.

Решение на языке C++

```
1  #include <iostream>
2  #include <cstdio>
3
4  using namespace std;
5
6  int main(){
7      freopen("input.txt", "r", stdin);
8      freopen("output.txt", "w", stdout);
9      int n;
10     cin>>n;
11     int mmm[26][26][26];
12     for(int i=0; i<n; ++i) for(int j=0; j<n; ++j) mmm[0][i][j]='Z'-i;
13     for(int i=0; i<n; ++i) for(int j=0; j<n; ++j) for(int k=1; k<n; ++k) mmm[k][j][i]='Z'-i;
14     for(int i=0; i<n; ++i){
15         for(int j=0; j<n; ++j){
16             for(int k=0; k<n; ++k) cout<<(char)mmm[i][j][k];
17             cout<<" ";
18         }
19         cout<<endl;
20     }
21     return 0;
22 }
```

Решение на языке Паскаль

```
1  var
```

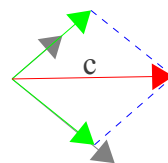
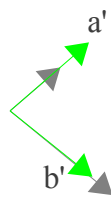
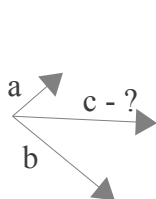
```
2   n, i, j, k : Longint;
3   mmm : array [0..25,0..25,0..25] of Longint;
4   begin
5     assign(input, 'input.txt'); reset(input);
6     assign(output, 'output.txt'); rewrite(output);
7
8     Read(n);
9     for i:=0 to n-1 do for j:=0 to n-1 do
10      mmm[0, i, j] := ord('Z') - i;
11     for i:=0 to n-1 do for j:=0 to n-1 do for k:=1 to n-1 do
12      mmm[k, j, i] := ord('Z') - i;
13     for i:=0 to n-1 do
14       for j:=0 to n-1 do begin
15         for k:=0 to n-1 do Write(chr(mmm[i, j, k]));
16         WriteLn;
17       end;
18   end.
```

Задача Е. Треугольник и окружности

Конечно же, для координат центров и радиусов вписанной и описанной окружностей имеются формулы, но их никто никогда не помнит. Поэтому нужно знать, что центр описанной окружности лежит в точке пересечения срединных перпендикуляров, а вписанной – в точке пересечения биссектрис; и уметь находить точки пересечения прямых.

В решениях ниже основную часть занимают формулы, поэтому не будем переносить их ещё и сюда. Поясним лишь нахождение биссектрисы угла.

Пусть у нас есть два вектора: \vec{a} и \vec{b} . Надо найти вектор \vec{c} , который будет коллинеарен биссектрисе угла, образованного векторами \vec{a} и \vec{b} .



Нормируем вектора

Зелёный треугольник равнобедренный найдем в нём медиану

$$\vec{a}' = \frac{\vec{a}}{|\vec{a}|}, \quad \vec{b}' = \frac{\vec{b}}{|\vec{b}|}$$

$$\vec{c} = \vec{a}' + \vec{b}'$$

Решение на языке C++

```
1 #include <iostream>
2 #include <cstdlib>
3 #include <cmath>
4
5 #define SQR(a) ((a)*(a))
6
7 using namespace std;
8
9 void bis(double x1, double y1, double x2, double y2, double &ax, double &ay){
10   double l1 = sqrt(x1*x1+y1*y1), l2 = sqrt(x2*x2+y2*y2);
11   ax = x1/l1+x2/l2;
12   ay = y1/l1+y2/l2;
13 }
14
15 void inter(double Ax, double Ay, double ax, double ay,
16            double Bx, double By, double bx, double by,
17            double &x, double &y){
18
19   double t = ((Bx-Ax)*by-(By-Ay)*bx)/(ax*by-ay*bx);
20   x = Ax + t * ax;
```

```
21     y = Ay + t * ay;
22 }
23
24 int main(){
25     freopen("input.txt","r",stdin);
26     freopen("output.txt","w",stdout);
27
28     double x[3], y[3], l[3];
29     for(int i=0;i<3;++i) cin>>x[i]>>y[i];
30     for(int i=0;i<3;++i) l[i] = sqrt(SQR(x[(i+1)%3]-x[i])+SQR(y[(i+1)%3]-y[i]));
31
32     char str[9];
33     cin>>str;
34
35     double ax, ay, bx, by;
36     bis(x[1]-x[0], y[1]-y[0], x[2]-x[0], y[2]-y[0], ax, ay);
37     bis(x[0]-x[1], y[0]-y[1], x[2]-x[1], y[2]-y[1], bx, by);
38
39     double x1, y1;
40     inter(x[0], y[0], ax, ay, x[1], y[1], bx, by, x1, y1);
41
42     if(str[0]=='I')
43         printf("%.6lf %.6lf %.6lf", x1, y1,
44             0.5*sqrt((l[0]+l[1]-l[2])*(l[1]+l[2]-l[0])*(l[2]+l[0]-l[1])/(l[0]+l[1]+l[2])));
45
46     ax = y[1]-y[0];
47     ay = x[0]-x[1];
48
49     bx = y[2]-y[1];
50     by = x[1]-x[2];
51
52     double x2, y2;
53     inter(0.5*(x[0]+x[1]), 0.5*(y[0]+y[1]), ax, ay, 0.5*(x[1]+x[2]), 0.5*(y[1]+y[2]), bx, by, x2, y2);
54
55     if(str[0]=='O') printf("%.6lf %.6lf %.6lf", x2, y2, sqrt(SQR(x2-x[0])+SQR(y2-y[0])));
56
57     return 0;
58 }
```

Решение на языке Паскаль

```
1 var
2     x, y, l : array [0..2] of Double;
3     i : Longint;
4     s : String;
5     ax, ay, bx, by, x1, y1, x2, y2 : Double;
6
7 procedure bis(x1, y1, x2, y2 : Double; var ax, ay : Double);
8 var
9     l1, l2 : Double;
10 begin
11     l1 := sqrt(sqr(x1)+sqr(y1));
12     l2 := sqrt(sqr(x2)+sqr(y2));
13     ax := x1/l1 + x2/l2;
14     ay := y1/l1 + y2/l2;
15 end;
16
17 procedure inter(Ax, Ay, _ax, _ay, Bx, By, _bx, _by : Double; var x, y : Double);
18 var
19     t : Double;
20 begin
21     t := ((Bx-Ax)*_by-(By-Ay)*_bx) / (_ax*_by-_ay*_bx);
22     x := Ax + t * _ax;
23     y := Ay + t * _ay;
24 end;
25
26 begin
27     assign(input, 'input.txt'); reset(input);
28     assign(output, 'output.txt'); rewrite(output);
29
30     for i:=0 to 2 do Read(x[i], y[i]);
31     for i:=0 to 2 do
32         l[i] := sqrt(sqr(x[(i+1) mod 3] - x[i]) + sqr(y[(i+1) mod 3] - y[i]));
```

```
33
34 ReadLn(s);
35 ReadLn(s);
36 bis(x[1]-x[0], y[1]-y[0], x[2]-x[0], y[2]-y[0], ax, ay);
37 bis(x[0]-x[1], y[0]-y[1], x[2]-x[1], y[2]-y[1], bx, by);
38
39 inter(x[0], y[0], ax, ay, x[1], y[1], bx, by, x1, y1);
40
41 if s[1] = 'I' then
    Write(x1:0:6, ' ', y1:0:6, ' ',
          (0.5*sqrt((l[0]+l[1]-l[2])*(l[1]+l[2]-l[0])*(l[2]+l[0]-l[1])/(l[0]+l[1]+l[2]))):0:6);
42
43 ax := y[1]-y[0];
44 ay := x[0]-x[1];
45
46 bx := y[2]-y[1];
47 by := x[1]-x[2];
48
49 inter(0.5*(x[0]+x[1]), 0.5*(y[0]+y[1]), ax, ay, 0.5*(x[1]+x[2]), 0.5*(y[1]+y[2]), bx, by, x2, y2);
50
51 if s[1] = 'O' then Write(x2:0:6, ' ', y2:0:6, ' ', sqrt(sqr(x2-x[0])+sqr(y2-y[0])):0:6);
52 end.
```