25ος ΠΑΝΕΛΛΗΝΙΟΣ ΔΙΑΓΩΝΙΣΜΟΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΘΕΜΑ Β΄ ΦΑΣΗΣ ΓΥΜΝΑΣΙΟΥ

Ενδεικτικές Λύσεις

PASCAL

Κουσουρής Γεώργιος Γυμνάσιο Κολεγίου Ψυχικού

```
program LimniKarla123;

VAR
   fin,fout:text;
   N,M,i,ii,jj,j,k,p:integer;
   karo:array[1..100,1..100] of integer;
   todo:array[1..2,1..1000] of integer;
   b:char;
 begin
   k:=0;
   assign(fin,'karla.in');
   reset(fin);
   read (fin,N);
   read (fin,b);
   readln (fin,M);

   for i:=1 to N do
     begin
       for j:=1 to N do
         begin
           read(fin,karo[i,j]);
           read(fin,b);
         end;
     end;
   close(fin);

  {for i:=1 to N do
     begin
       for j:=1 to N do
         begin
           if karo[i,j]>M then
             karo[i,j]:=-1
           else
             karo[i,j]:=-2;
         end;
     end;
   }
for i:=1 to N do
  begin
```

```
    for j:=1 to N do
          begin
          if karo[i,j]>M then
            begin
             k:=k-1;
             ii:=i;
             jj:=j;
             p:=0;
             repeat
                 if P>=1 then
                   begin
                     ii:=todo[1,p];
                     jj:=todo[2,p];
                     p:=p-1;
                   end;

                     if  (jj<>N) and (karo[ii,jj+1]>M) then
                        begin
                          p:=p+1;
                          todo[1,p]:=ii;
                          todo[2,p]:=jj+1;
                        end;

                     if (ii<>N) and (karo[ii+1,jj]>M) then
                       begin
                          p:=p+1;
                          todo[1,p]:=ii+1;
                          todo[2,p]:=jj;
                       end;

                     if (jj<>1) and (karo[ii,jj-1]>M) then
                        begin
                          p:=p+1;
                          todo[1,p]:=ii;
                          todo[2,p]:=jj-1;
                        end;
                     if (ii<>1) and (karo[ii-1,jj]>M) then
                       begin
                          p:=p+1;
                          todo[1,p]:=ii-1;
                          todo[2,p]:=jj;
                       end;

                     karo[ii,jj]:=k;
              until P=0;
            end;
    end;
 end;
  assign(fout,'karla.out');
  rewrite(fout);
```

```
   writeln (fout,ABS(k));
   writeln (ABS (k));
   close (fout);
end.
```

C

Σταματίου Γεώργιος Γυμνάσιο Αμερικανικού Κολεγίου Ελλάδος

```
#include <stdio.h>

#define MAX_DIM                  101

//Variable Declerations
int table[MAX_DIM + 1][MAX_DIM + 1];
short visited[MAX_DIM + 1][MAX_DIM + 1];
int N, M, K;

int flood(int a, int b)     {
      //Out of bounds check
      if(   a == 0      ||
            a > N ||
            b == 0      ||
            b > N ||
            visited[a][b] == 1 ||
            table[a][b] <= M)
            return 0;

      visited[a][b] = 1;
      flood(a + 1, b);
      flood(a - 1, b);
      flood(a, b + 1);
      flood(a, b - 1);

      return 1;
}
int main(void)   {
      FILE *in = fopen("karla.in", "rt"), *out =
fopen("karla.out", "wt");

      //Input proceudure
      fscanf(in, "%d%d", &N, &M);
      int i , j;
      for(i = 1; i <= N; i++)     {
            for(j = 1; j <= N; j++)
                  fscanf(in, "%d", &table[i][j]);
      }
```

```
      //Processing
      for(i = 1; i <= N; i++)     {
            for(j = 1; j <= N; j++)     {
                  if(flood(i, j) == 1)
                        K++;
            }
      }

      fprintf(out, "%d", K);

      return 0;
}
```

C++

Χαλδέζος Ιωάννης **5° Γυμνάσιο Μυτιλήνης**

```
#include <iostream>
#include <fstream>

using namespace std;
bool lake[102][102];
int left1;

void flood(int a, int b)
{
    if (lake[a][b])
    {
        lake[a][b]=false;
        left1--;

        flood(a-1,b);
        flood(a+1,b);
        flood(a,b-1);
        flood(a,b+1);

    }
}

int main()
{
   int n,high,i,j,temp,cnt=0;

   ifstream data("karla.in");
   data>>n;
   data>>high;
   for (i=0;i<=n+1;i++)
    {
```

```
        lake[i][0]=false;
        lake[0][i]=false;
        lake[n+1][i]=false;
        lake[i][n+1]=false;
    }


    left1=n*n;
   for (i=1;i<=n;i++)
      {
      for (j=1;j<=n;j++)
        {
          data>>temp;
          if (temp>high) lake[i][j]=true;
          else {lake[i][j]=false; left1--;}
        }
      }
     cnt=0;

     while (left1>0)
     {
    for (i=1;i<=n;i++)
        for (j=1;j<=n;j++)
            if (lake[i][j]){ flood(i,j);cnt++;}
    }
    ofstream outdata("karla.out");
    outdata<<cnt;
    return 0;
}
```