24ος ΠΑΝΕΛΛΗΝΙΟΣ ΔΙΑΓΩΝΙΣΜΟΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΑΠΑΝΤΗΣΕΙΣ Β΄ ΦΑΣΗΣ ΛΥΚΕΙΟΥ

Ραδιοαστέρες (Pulsars)

C++

ΚΑΝΕΛΗΣ ΚΩΝΣΤΑΝΤΙΝΟΣ (2ο ΓΕΛ ΘΕΣΣΑΛΟΝΙΚΗΣ)

```
/* NAME: KONSTANDINOS KANELIS
 * TASK: PULSARS
 * LANG: C++
 */
#include<stdio.h>
#include<algorithm>
#define MAXN 1000010
using namespace std;
struct pt{
	int x,y,id;
	bool operator <(const pt& a) const{
		return (x < a.x) || (x == a.x && y < a.y);
	}
};
pt P[MAXN],C[2*MAXN];
int N,M=0;

int CCW(pt a,pt b,pt c){
	return (b.x-a.x)*(c.y-a.y) - (b.y-a.y)*(c.x-a.x);
}
bool cmp(pt a,pt b){
	return a.id<b.id;
}
int main()
```

```
{
        FILE* fin = fopen("pulsars.in","r");
        FILE* fout = fopen("pulsars.out","w");
        int i,t;

        fscanf(fin,"%d",&N);
        for(i=0;i<N;i++){
                fscanf(fin,"%d %d",&P[i].x,&P[i].y);
                P[i].id = i+1;
        }
        sort(P,P+N);
        for(i=0;i<N;i++){
                while(M >=2 && CCW(C[M-2],C[M-1],P[i]) <= 0) M--;
                C[M++] = P[i];
        }
        for(i=N-2,t=M+1; i>=0; i--){
                while(M >=t && CCW(C[M-2],C[M-1],P[i]) <= 0) M--;
                C[M++] = P[i];
        }
        M--;
        sort(C,C+M,cmp);
        fprintf(fout,"%d\n",M);
        for(i=0;i<M;i++){
                fprintf(fout,"%d\n",C[i].id);
        }
        fclose(fout);
        return 0;
}
}
```

C

ΣΑΚΚΑΣ ΓΕΩΡΓΙΟΣ (4ο ΓΕΛ Καλαμάτας)

```
/* NAME: GEORGIOS SAKAS
 * TASK: PULSARS
 * LANG: C
 */
#include <stdio.h>
#include <stdlib.h>
int main(void)
{
 int shmeia,i;
 FILE *fin=fopen("pulsars.in","r");
 fscanf(fin,"%d", &shmeia);

 int *x, *y;
 x=(int*)calloc(shmeia,sizeof(int));
 y=(int*)calloc(shmeia,sizeof(int));
 for(i=0;i<shmeia;i++) fscanf(fin,"%d %d", &x[i], &y[i]);
 fclose(fin);

 int min=y[0],minth=0;
 for(i=1;i<shmeia;i++)
 {
    if(min>y[i])
    {
       min=y[i];
       minth=i;
    }
 }
 int max=y[0],maxth=0;
 for(i=1;i<shmeia;i++)
 {
    if(max<y[i])
    {
       max=y[i];
       maxth=i;
    }
 }

 int *kor, shm=minth,count=0,blocks=50;
 kor=(int*)calloc(blocks,sizeof(int));
 kor[0]=minth;
 while(shm!=maxth)
 {
    shm=maxth;
    for(i=0;i<shmeia;i++)
```

```
    {
       if(i!=kor[count] && (x[shm]-x[kor[count]])*(y[i]-
y[kor[count]])-(y[shm]-y[kor[count]])*(x[i]-x[kor[count]])<0)
shm=i;
    }
    count++;
    if(count>=blocks)
    {
       blocks+=10;
       kor=(int*)realloc(kor,blocks*sizeof(int));
    }
    kor[count]=shm;
 }
 shm=maxth;
 while(shm!=minth)
 {
    shm=minth;
    for(i=0;i<shmeia;i++)
    {
       if(i!=kor[count] && (x[shm]-x[kor[count]])*(y[i]-
y[kor[count]])-(y[shm]-y[kor[count]])*(x[i]-x[kor[count]])<0)
shm=i;
    }
    count++;
    if(count>=blocks)
    {
       blocks+=10;
       kor=(int*)realloc(kor,blocks*sizeof(int));
    }
    kor[count]=shm;
 }
 free(x);
 free(y);

 FILE *fout=fopen("pulsars.out","w");
 fprintf(fout,"%d\n", count);
 for(i=0;i<=shmeia;i++)
 {
    int j=0;
    while(i!=kor[j] && j<count) j++;
    if(i==kor[j]) fprintf(fout,"%d\n", kor[j]+1);
 }
 fclose(fout);
 free(kor);

 return(0);
}
```

PASCAL

ΜΑΡΑΓΚΟΥ ΣΟΦΙΑ (ΓΕΛ ΣΥΡΟΥ)

```
(* NAME: SOFIA MARAGOU
 * TASK: PULSARS
 * LANG: PASCAL
*)
program pdp_pulsars;
var userfile:text;
    st:string;
    location,error:integer;
    s:array[1..1000000,1..2] of word;
    k:array[1..1000000] of longint;
    z,b,x1,y1,a,us,d,i,max,min,j,n:longint;
    bmax,flag:boolean;
    smax,sy:real;
    temp:longint;

begin
    Assign(userfile,'pulsars.in');
    reset(userfile);
    readln(userfile,st);
    val(st,n,error);

    for i:=1 to n do
    begin
       readln(userfile,st);
       location:=pos(' ', st);
       val(copy(st,1,location-1),s[i,1],error);
       val(copy(st,location+1,length(st)-location),s[i,2],er-
ror);

    end;
    close(userfile);

    max:=1;min:=1;
    for i:=1 to n do
    begin
       if s[i,2] > s[max,2] then
         max:=i
       else
         begin
           if s[i,2]=s[max,2] then
             if s[i,1]>s[max,1] then max:=i;
         end; {else}

        if s[i,2]<s[min,2] then
```

```
            min:=i
         else
            begin
               if s[i,2]=s[min,2] then
                 if s[i,1]>s[min,1] then min:=i;
             end; {else}
    end; {for}
    for i:=1 to n do k[i]:=0;

    d:=1; k[d]:=min; bmax:=false; flag:=false; us:=min;
smax:=-2;
    while (flag=false) do
    begin
       if bmax=false then
       begin
          smax:=-2;
          for a:=1 to n do
          begin
            if a<>us then
              if s[a,2]>=s[us,2] then
              begin
                 x1:=(s[a,1]-s[us,1]);
                 y1:=(s[a,2]-s[us,2]);
                 sy:=x1/sqrt(sqr(x1)+sqr(y1));
                 if sy>smax then
                 begin
                    smax:=sy; b:=a;
                 end;{if}
              end;{if}
          end; {for}

          us:=b; d:=d+1; k[d]:=us;
          if us=max then bmax:=true;
       end {if}
       else
       begin
          smax:=2;
          for a:=1 to n do
          begin
            if a<>us then
              if s[a,2]<=s[us,2] then
              begin
                 x1:=(s[a,1]-s[us,1]);
                 y1:=(s[a,2]-s[us,2]);
                 sy:=x1/sqrt(sqr(x1)+sqr(y1));
                 if sy<smax then
                 begin
                    smax:=sy; b:=a;
```

```
                end;{if}
              end;{if}
          end; {for}

          us:=b;
          if us=min then
            flag:=true
          else
            begin
               d:=d+1;
               k[d]:=us;
            end;
        end ;{else}
    end; {while}

    for z:=2 to d do
    begin
      for i:=d downto z  do
        if k[i]<k[i-1] then
        begin
          temp:=k[i]; k[i]:=k[i-1];k[i-1]:=temp;
        end;
    end;


    assign(userfile,'pulsars.out');
    rewrite(userfile);
    writeln(userfile,d);
    for i:=1 to d do writeln(userfile,k[i]);
    close(userfile);

end.
```

Οι παραπάνω απαντήσεις είναι ενδεικτικές και κατά συνέπεια δεν σημαίνει ότι και άλλοι μαθητές δεν υπέβαλαν αντίστοιχες λύσεις.