27^{ΟΣ} ΠΑΝΕΛΛΗΝΙΟΣ ΔΙΑΓΩΝΙΣΜΟΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ Α΄ ΦΑΣΗΣ

«Το κυλικείο του σχολείου»

PASCAL

ΠΑΛΑΙΟΛΟΓΟΣ ΔΙΑΣ   **(εκτός συναγωνισμού Δημ. Σχ. Μυτιλήνης)**

```pascal
program ypsos;
var
 board:array [1..1000000] of longint;
 N,max,i,counter:longint;
 infile,outfile:text;
begin
 assign(infile,'xxx.in');
 reset(infile);
 readln(infile,N);
 for i:=1 to N do
   read(infile,board[i]);

 max :=board[N];

 counter:=1;
  for i:=N-1 downto 1 do
   begin
    if board[i]>max then
     begin
      max:=board[i];
      counter:=counter+1;
     end;
   end;
 assign(outfile,'xxx.out');
 rewrite (outfile);
 writeln(outfile,counter);
 close(infile);
 close(outfile);
end.
```

C

ΑΛΒΕΡΤΟΣ-ΑΘΑΝΑΣΙΟΣ ΖΑΦΕΤ   ΕΕΙ-ΚΟΛΛΕΓΙΟ ΨΥΧΙΚΟΥ

```c
#include<stdio.h>
#include<stdlib.h>
int main()
{
    FILE *fd;
    fd = fopen("xxx.in","r");
    freopen("xxx.out","w",stdout);

    int N,*Kyl;
    fscanf(fd,"%d",&N);
    Kyl = (int *)malloc(N*sizeof(int));


    int i=0;
    while (i<N) {
        fscanf(fd, "%d", &Kyl[i]);
        i=i+1;
    }

    int p=1;
    int max=Kyl[N-1];
    i=N-2;
    while (i>=0) {
     if (Kyl[i]>max) {
       max=Kyl[i];
       p=p+1;
     }
    i=i-1;
    }

    printf("%d\n", p);
    return 0;
}
```

CPP

ΤΣΑΜΗΣ ΕΜΜΑΝΟΥΗΛ   ΙΔ. ΓΕΛ. ΕΛΛΗΝΙΚΟ ΚΟΛΛΕΓΙΟ ΘΕΣ/ΙΚΗΣ

```c
#include <stdio.h>
#include <stdlib.h>

FILE *in, *out;

int main(int argc, char** argv) {
        in = fopen("xxx.in", "r");
        int total;
    int size;
    int offset = 0;
        fscanf(in, "%d", &total);

    fseek(in, 0L, SEEK_END);

    #if defined(_WIN32)
            size = ftell(in) - 1;
    #elif defined(_WIN64)
            size = ftell(in) - 1;
        #else
            size = ftell(in);
        #endif

        fseek(in, 0L, SEEK_SET);

    char *buffer = (char*) malloc(size + 1);
    char *token;

    fread(buffer, 1, size, in);
    buffer[size] = '\n';

    int *values = (int*) malloc(total * sizeof(int));
    int i;

    while(*buffer != '\n'){
            buffer++;
    }

    buffer++;
```

```
    for (i=0;i<total;i++){
       values[i] = 0;

       while(*buffer != ' ' && *buffer != '\n'){
                       values[i] = (values[i] * 10) + (*buffer - '0');
          buffer++;
          }

          buffer++;
    }

    int maxValue = values[total - 1];
    int output = 1;

    for (i=total-2;i>=0;i--) {
        if (values[i] > maxValue) {
                maxValue = values[i];
                output++;
        }
        }

        fclose(in);

        out = fopen("xxx.out", "w");
    fprintf(out, "%d", output);

    fclose(out);

        return 0;
}
```