

АНАЛИЗ НА ЗАДАЧА КУЛИ

Основната идея на решението е по време на изпълнението на заявките да помним броя на двойките съседни числа, които не са подредени в ненамаляващ ред. Така цялата редица ще бъде в исканата наредба, тогава и само тогава, когато този брой е равен на 0.

В началото изчисляваме броя, сравнявайки всеки две съседни кули. При постъпване на заявка за промяна на височината на дадена кула е достатъчно да разгледаме отношението и с двете съседни кули. Проверяваме поотделно за лявата и дясната (ако такива съществуват) дали от неправилна наредба преминават в правилна или обратното и привеждаме поддържащия брой в съответствие.

Състезателите трябва да използват бърз вход и изход, за да получат 100 точки. Първият вариант е да ускорят работата на стандартните потоци за вход и изход **cin** и **cout** като добавят следните редове в началото на програмите си:

```
ios :: sync_with_stdio(false);  
cin.tie(NULL);
```

Важно е да се отбележи, че манипулаторът **endl** също забавя изпълнението на програмата, и поради това следва да бъде заменен с извеждане на символ за нов ред – ‘\n’.

Вторият вариант, който е малко по-бавен първия, е да се използват функциите за вход и изход от библиотеката **cstdio** – **scanf** и **printf**.