

Bantumi 4

Questo esercizio è facoltativo, ma chi decide di farlo **deve consegnarlo entro il 1/7/2015**. Alla consegna il vostro programma viene fatto giocare contro il mio. Se il vostro programma vince, passa la prima fase. Nella seconda fase i programmi che hanno passato la prima fase vengono fatti giocare tra loro. Verrà stilata una classifica dei migliori programmi, in funzione del numero di vittorie conseguite. Questi programmi verranno anche visionati da me e la loro chiarezza e semplicità avrà un peso sul giudizio finale. Ai migliori programmi verranno attribuiti al massimo 3 punti. Chi passa le eliminatorie avrà al massimo 2 punti. Chi consegna un programma che vince qualche partita contro il mio avrà al massimo 1 punto. I punti conseguiti verranno aggiunti a quelli dell'esame (che deve essere già sufficiente)

L'esercizio richiede di realizzare una funzione che giochi a Bantumi in modo "intelligente". Questa funzione deve soddisfare la seguente specifica:

PRE=(B rappresenta la situazione del gioco, player=0/1, livello>=0)

int meglio_mossa(int B[][7],int player,int livello)

POST=(restituisce una mossa m per player, $0 \leq m \leq 5$ e $M[\text{player}][m] \neq 0$)

Il vostro programma deve stare su un file che si chiama EXTRA.cpp. Dopo le solite istruzioni

```
#include<iostream>
#include<fstream>
using namespace std;
```

il vostro programma deve contenere la dichiarazione :

```
namespace EXTRA{
```

```
//qui va inserito tutto quello che serve alla vostra funzione meglio_mossa
```

```
}//questa è l'ultima riga del programma
```

Il vostro programma non deve contenere un main. Quando consegnate il vostro file nell'esercizio Bantumi 4 del moodle esso verrà fatto giocare contro il mio programma. L'output mostrerà il risultato di queste partite specificando quante partite ha vinto il giocatore 0 (io) e quante ne ha vinte il giocatore 1 (voi).

Per mostrare questo output, sfruttiamo il test automatico dei programmi che in questo caso non ci serve visto che comunque l'output (cioè il risultato delle partite) non è prevedibile. Quindi stabilisco un output atteso di fantasia, per esempio, "vediamo chi vince!!" che sarà sempre diverso da quello prodotto dal programma e quindi l'output del programma vi verrà mostrato come fosse un errore. Ogni mossa prodotta dalla vostra funzione meglio_mossa viene controllata dal mio programma e in caso la mossa sia sballata (o fuori da [0..5] o indichi una buca vuota) viene prodotto un output che segnala l'errore e il programma termina.

Come già detto prima, le consegne che battono il mio programma verranno fatte giocare tra loro. Per questa seconda fase è sufficiente la consegna del Bantumi 4, non dovrete fare nulla più di questo. I risultati finali verranno pubblicati su questo moodle.

