

Vorlesung
EPROG

ESC 2015 - Eprog Student Challenge

Ziel der ESC 2015 ist es, einen Bot zu programmieren, der *4-gewinnt* spielen kann. Die Bots der Teilnehmer werden dann gegeneinander antreten und so einen Sieger bestimmen. (siehe auch de.wikipedia.org/wiki/Vier_gewinnt) Es gibt 100€ Siegprämie (und immerwährenden Ruhm) und 50€ für den zweiten Platz.

Teilnahme

Jeder EPROG Teilnehmer, der bis Ende März den Quelltext eines Bots (genauere Erklärung weiter unten) an michael.feischl@tuwien.ac.at schickt, nimmt an der ESC teil.

Wie sollen diese Bots aussehen

Die Programmiersprache für diesen Wettbewerb ist C. Der Quelltext soll in einer `.c` Datei abgegeben werden, die

- kein `main()`-Programm enthält,
- eine Funktion mit der Signatur

```
int makeMove_NACHNAME(int myself, int opponent, int** board, int last)
```

enthält (keine Umlaute oder Sonderzeichen),

- beliebig viele weitere (Hilfs-)Funktionen und alle nötigen `#include` enthält. Um Konflikte zwischen den Bots zu vermeiden, muss jeder weitere Funktionsname mit `_NACHNAME` enden.

Was sollen diese Bots machen

Die Funktion `makeMove_NACHNAME(...)` bekommt als Übergabeparameter das Spielfeld `board` welches ein 6×7 -Array vom Typ `int**` ist. Dabei gelten folgende Regeln

5								
4								
3								
2								
1								
0								
j/k	0	1	2	3	4	5	6	

`board[j][k] = 0` ... leeres Feld,
`board[j][k] = myself` ... eigener Stein,
`board[j][k] = opponent` ... gegnerischer Stein.

Der Integer `last` $\in \{0, \dots, 6\}$ gibt jene Spalte an, in die der Gegner seinen letzten Stein geworfen hat. Beim ersten Zug ist `last = -1`. Die Funktion soll nun einen *gültigen* Zug machen, d.h. einen Integer `k` $\in \{0, \dots, 6\}$ zurückgeben, der die (nicht volle) Spalte angibt, in die der eigene Stein geworfen werden soll. **Wichtig:** Das Spielfeld `board` darf nicht verändert werden

(Disqualifikation!). Es darf zusätzlicher Speicher angelegt werden, muss aber nach dem Ende der Funktion `makeMove_NACHNAME(...)` wieder freigegeben werden. Speicherlecks resultieren in Punktabzug.

Wie läuft das ganze ab

Ein Duell zwischen den Bots von hypothetischen EPROG Studenten Alice und Bob läuft wie folgt ab (das Duellprogramm wird natürlich von den Organisatoren erstellt):

1. Erstelle `int** board` und fülle es mit Nullen (leeres Feld). Setze `last=-1`.
2. `last=makeMove_ALICE(1,2,board,last)`
3. Werfe Stein von Alice in Spalte `last`.
4. Hat jemand gewonnen oder unentschieden? Wenn nein, dann gehe zu 5.
5. `last=makeMove_BOB(2,1,board,last)`
6. Werfe Stein von Bob in Spalte `last`.
7. Hat jemand gewonnen oder unentschieden? Wenn nein gehe zu 2.

Ein ungültiger Zug (Spalte bereits voll) resultiert in einer Niederlage für den ziehenden Spieler. Dauert ein Spiel länger als zwei Minuten, verliert automatisch jener Spieler, der bis dahin die meiste Rechenzeit verbraucht hat. Gewertet wird im Bundesligasystem: Jedes Gegnerpaar hat zwei Spiele, wobei jeder einmal beginnen darf. Für einen Sieg gibt es 3 Punkte, Unentschieden 1 Punkt, Niederlage 0 Punkte. Bei gleicher Punktezahl entscheiden zuerst die Anzahl der gewonnenen Partien und dann die insgesamt verbrauchte Rechenzeit.

Beispiel

Ein mögliches Spiel zwischen Alice (1) und Bob (2) könnte wie folgt aussehen (über den Pfeilen stehen die Rückgabewerte der Funktionen `makeMove_ALICE` und `makeMove_BOB`).

