hochschule mannheim

Fakultät für Informatik
Prof. Dr. Peter Knauber

Bachelor-Studiengang Informatik
Übungen zur Vorlesung “Grundlagen der Informatik” (GDI), WS 2011/2012

Übungsblatt 11: Klassen, Arrays

Ausgabe am: 9.12.2011
Abgabe am: 23.12.2011

Aufgabe 1: **25 Punkte**

Erstellen Sie eine Implementierung für das Spiel “Vier gewinnt” in Form eines Java-Programms. Bei “Vier gewinnt” spielen zwei Spieler gegeneinander. Das Spielbrett steht dabei senkrecht und ist hohl. Von oben können Spielsteine eingeworfen werden, die dann an die unterste freie Position rutschen und dort liegenbleiben. Die Spieler lassen nacheinander jeweils einen Spielstein in das Spielbrett fallen. Ziel des Spieles ist es eine Reihe von vier eigenen Steinen horizontal, vertikal oder diagonal anzuordnen. Das Spielbrett ist üblicherweise sieben Felder breit und sechs Felder hoch. Weitere Informationen: de.wikipedia.org/wiki/Vier_gewinnt

Erläuterungen zum Spielfeld

Benutzen sie für die Implementierung ein statisches zweidimensionales Array des Typs *char*. Jedes Feld in diesem Array soll entweder ein Leerzeichen enthalten, wenn es leer ist, oder einen Spielstein in Form eines ‘X’ für den ersten Spieler bzw. eines ‘O’ für den zweiten Spieler.
Ein zweidimensionales Array mit Breite x = 7 und Höhe y = 6 kann man sich so vorstellen:

0,0	1,0	2,0	3,0	4,0	5,0	6,0
0,1	1,1	2,1	3,1	4,1	5,1	6,1
0,2	1,2	2,2	3,2	4,2	5,2	6,2
0,3	1,3	2,3	3,3	4,3	5,3	6,3
0,4	1,4	2,4	3,4	4,4	5,4	6,4
0,5	1,5	2,5	3,5	4,5	5,5	6,5

Die Indizes stehen in dem jeweiligen Feld in der Form “x,y”.
(Hinweis: In welche Richtung sich die x-Achse und y-Achse ausbreiten ist reine Interpretationssache.)

Das zweidimensionale Spielfeld kann in Java wie folgt vereinbart werden:

```
char[][] board = new char[7][6];
```

Erklärung:

- *char[][]* bezeichnet einen Typ, bei dem ein Array andere Arrays vom Type *char* beinhaltet, vgl. Abbildung.
- *new char[7][6]* erstellt ein solches zweidimensionales Array mit der Breite 7 und der Höhe 6.
- Für den Zugriff auf ein zweidimensionales Array braucht man zwei Indizes: *board[x][y]*; ‘x’ bezieht sich auf die Spalte und ‘y’ auf die Reihe.

Vorgaben für die Implementierung

- Eine *main*-Methode bietet ein Menü an, um ein neues Spiel zu starten oder das Programm zu beenden. Das Spiel selbst wird in einer Schleife realisiert, in der die Spieler abwechselnd ihren Zug machen können.
- Eine Methode *init()* setzt alle Felder des Spielbretts auf ‘ ‘ (Leerzeichen). Diese Methode muss vor jedem Spiel ausgeführt werden, um sicher zu stellen, dass alle Felder des zweidimensionalen Arrays korrekt initialisiert sind.
- Eine Methode
  boolean putToken(int column, char token)
  fügt den angegebenen Spielstein *token* für den aktuellen Spieler in die angegebene Spalte *column* (0 - 6). Die Methode gibt *true* zurück, wenn der Spielstein erfolgreich hinzugefügt wurde. Sie gibt *false* zurück, wenn der Einwurf fehlgeschlagen ist, d.h. wenn die Spalte bereits voll ist oder wenn der Index ungültig ist.
- Eine Methode
  char checkVictory()
  prüft das Spielbrett daraufhin, ob ein Spieler gewonnen hat. Sie gibt entweder das Zeichen des Spielers zurück, der gewonnen hat, oder ein Leerzeichen, wenn kein Spieler gewonnen hat.
- Eine Methode
  boolean checkDraw()
  prüft das Spielbrett und liefert *true* zurück, wenn das Spiel unentschieden ausgegangen ist, sonst *false*.

Aufgabenstellung

Implementieren Sie das Spiel nach diesen Vorgaben. Bereiten Sie sich darauf vor, Ihr Spiel zu erklären (ohne Erklärung keine Punkte!). Denken Sie vor allem an Erklärungen, warum Sie welchen Klassen, Methoden etc. welche Aufgaben übertragen haben.

Tipps zur Implementierung

- Schreiben Sie eine Methode, die das komplette Spielfeld auf der Konsole ausgibt, und die Sie nach jedem Zug aufrufen.
- Geschickt wäre eine Klasse *Spielfeld*, die das zweidimensionale Array verwaltet und einige der geforderten (welche?) Methoden dafür zur Verfügung stellt.
- Zum Prüfen, ob ein Spieler gewonnen hat, eignen sich drei Hilfsmethoden:
  - *char checkVertical()*
    Sucht nach vertikalen Folgen von 4 gleichen Spielstein.
  - *char checkHorizontal()*
    Sucht nach horizontalen Folgen von 4 gleichen Spielstein.
  - *char checkDiagonal()*
    Sucht nach diagonalen Folgen von 4 gleichen Spielstein.

Zusatzaufgabe 2 **10 Punkte**

Jeder Spieler soll die (komfortable) Möglichkeit haben, das Spiel im aktuellen Zustand zu speichern, um es zu einem späteren Zeitpunkt wieder aufzunehmen. Dazu ist das aktuelle Spielfeld zu speichern sowie die Information, welcher Spieler gerade am Zug ist.

Gleichzeitig ist das Hauptmenü zu erweitern, so dass ein gespeichertes Spiel wieder geladen werden kann.

Zusatzaufgabe 3 **15 Punkte**

Erstellen Sie eine Programmversion, in der Sie gegen den Computer spielen können. Tipps:

- Hilfreich ist eine Methode
  void removeToken(int column),
  die einen Spielstein aus der Spalte mit dem Index *column* (0 - 6) entfernt.
- Erster Schritt: Benutzen Sie die Methoden *putToken, checkVictory* und *removeToken*, um dann, wenn der Computer am Zug ist, nacheinander die Spalten daraufhin zu prüfen, ob er durch Einwerfen eines Spielsteins gewinnen kann.
- Zweiter Schritt: Wenn kein direkter Gewinn für den Computer möglich ist, benutzen Sie die gleichen Methoden, um festzustellen, ob der menschliche Spieler durch Einwerfen in eine bestimmte Spalte gewinnen kann: Dann sollte der Computer diese Spalte durch einen eigenen Spielstein blockieren.
- Dritter Schritt: …?

Hinweise

- Vergeben Sie vernünftige Namen für Ihre Variablen und Parameter und vergessen Sie die Kommentare nicht!
- Die Aufgabe ist in Eclipse zu bearbeiten, legen Sie dafür ein Package *uebung11* an.
- Von der Aufgabe ist ein Programmausdruck (Listings) abzugeben, *keine* Ausdrucke von Testläufen. Die Aufgaben sind im Labor mit Eclipse vorzuführen.
- Erlaubt sind *MakeItSimple*-Funktionen (keine nicht besprochene Funktionalität aus der Java Standard Bibliothek) und das bisher erworbene Wissen aus den GDI-Vorlesungen. Fragen Sie, bevor Sie Java-Konstrukte verwenden, die noch nicht behandelt wurden! Zusätzliche eigene Hilfsfunktionen (keine fremden oder externen) sind ausdrücklich erlaubt.