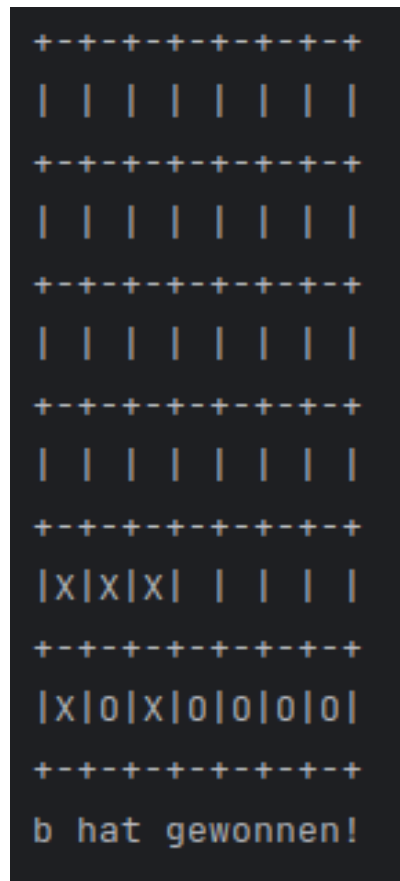


Vier Gewinnt – Spiel



Autor:

Wolfgang Kremer

wolfgang.kremer@tn.mischok.academy

Im Auftrag von:

Mischok Academy

Steingasse 13

86150 Augsburg

Inhaltsverzeichnis

Projektbeschreibung.....	3
Verwendete Technik.....	3
Hauptkomponenten.....	3
'Main' Klasse.....	3
Tic-Tac-Toe Spiel.....	3
Projektbeschreibung.....	3
Verwendete Technologien.....	3
Hauptkomponenten.....	4
Main Klasse.....	4
'printBoard' Methode.....	4
'playerOneMove' und playerTwoMove Methoden.....	4
'placeMove' Methode.....	4
'isValidMove' Methode.....	4
'isGameFinished' Methode.....	4
'wins' Methode.....	4
Anwendung.....	4
Mögliche Änderungen und Verbesserungen.....	5
Weitere Dokumentationen des Projekts.....	5

Projektbeschreibung

Dieses Projekt implementiert ein einfaches Vier Gewinnt – Spiel für zwei Spieler. Es verwendet die Konsole als Benutzeroberfläche. Jeder Spieler kann nacheinander seinen Zug machen bis entweder ein Sieger ermittelt wurde oder das Spielfeld voll ist. In diesem Fall endet das Spiel Unentschieden.

Verwendete Technik

- Java
- Scanner (zur Benutzereingabe)

Hauptkomponenten

‘Main’ Klasse

- Die Hauptklasse, die das Spiel steuert und die Benutzereingabe verarbeitet.
- Initialisiert das Spielfeld als zweidimensionales char-Array (7x6).
- Liest die Namen der beiden Spieler ein.
- Enthält die Spiellogik und die Schleifen, um die Runden der Spieler zu wiederholen.

Eine Dokumentation für dieses Projekt sollte klar und präzise sein, um anderen Entwicklern oder Personen, die das Projekt verwenden oder verstehen möchten, zu helfen. Hier ist eine Beispielstruktur für die Dokumentation:

Tic-Tac-Toe Spiel

Projektbeschreibung

Dieses Projekt implementiert ein einfaches Tic-Tac-Toe-Spiel für zwei Spieler. Es verwendet die Konsole als Benutzeroberfläche und ermöglicht den Spielern, abwechselnd ihre Züge zu machen, bis einer der Spieler gewinnt oder das Spiel unentschieden endet.

Verwendete Technologien

- Java
- Scanner (für Benutzereingaben)

Hauptkomponenten

Main Klasse

- Die Hauptklasse, die das Spiel steuert und die Benutzereingabe verarbeitet.
- Initialisiert das Spielfeld als zweidimensionales char-Array.
- Liest die Namen der beiden Spieler ein.
- Enthält die Spiellogik und die Schleifen, um die Runden der Spieler zu wiederholen.

‘printBoard’ Methode

- Zeigt das Spielfeld in der Konsole an.

‘playerOneMove’ und playerTwoMove Methoden

- Verarbeiten die Züge der Spieler eins und zwei.
- Überprüfen die Gültigkeit der Züge und aktualisieren das Spielfeld.

‘placeMove’ Methode

- Setzt den Spielstein (X oder O) an der gewünschten Position auf dem Spielfeld.

‘isValidMove’ Methode

- Überprüft, ob ein Zug an einer bestimmten Position gültig ist.

‘isGameFinished’ Methode

- Überprüft, ob das Spiel beendet wurde (ein Spieler hat gewonnen oder es ist unentschieden).

‘wins’ Methode

- Überprüft, ob ein Spieler gewonnen hat.

Anwendung

Um das Spiel zu spielen, führen Sie die Main Klasse aus und folgen Sie den Anweisungen zur Eingabe der Spieler-Namen und ihrer Züge.

- Spieler 1 Namenseingabe
- Spieler 2 Namenseingabe
- Spieler 1 Zug
- Spieler 2 Zug

Mögliche Änderungen und Verbesserungen

- Implementierung einer graphischen Benutzeroberfläche (GUI)
- Hinzufügen einer Möglichkeit das Spiel neu zu starten
- Hinzufügen eines Zählers (wie oft hat ein Spieler in der jeweiligen Session gewonnen)

Weitere Dokumentationen des Projekts

- Pseudocode in der pseudocode.txt