

Aufgabe 3: Tobis Turnier

Team-ID: 00523

Team: Einstein One

Bearbeiter/-innen dieser Aufgabe:

Martin Nikoljacic

21. November 2020

Inhaltsverzeichnis

Lösungsidee	1
Umsetzung	1
Beispiele.....	3
Quellcode	5

Lösungsidee

Die Grundidee bei Tobis Turnier ist folgende:

Aus der vorgegebenen Datei wird die Anzahl der Spieler, sowie die verschiedenen Spielerstärken entnommen.

Aus diesen Informationen wird dann eine Spielerliste angelegt.

Danach werden alle drei Turnier-Varianten ausprobiert und die gesamten Siege des besten Spielers pro Turnier vermerkt.

Die durchschnittlichen Siege in den angegebenen Wiederholungen werden dann verglichen. Das Turnier-System, bei dem der beste Spieler durchschnittlich am häufigsten gewinnt wird ausgegeben.

Turnier Varianten (Siehe auch Beschreibung unter Materialien):

Liga:

Jeder Spieler soll einmal gegen alle anderen gespielt haben.

KO-System:

Die Spieler werden gemäß eines Turnierplans in Paare eingeteilt. Jedes Paar spielt einmal gegeneinander. Der Sieger kommt in die nächste Runde

KO5-System:

Das Prinzip ist das gleiche, wie beim KO-System, nur wird der Gewinner einer Partie ermittelt, indem die beiden Spieler fünf mal Gegeneinander spielen.

Umsetzung

Zur Umsetzung der Lösungsidee, in Java, muss zuerst eine Datei des entsprechenden Formats eingelesen werden. Dazu können FileChooser und FileReader verwendet werden, welche erst eine

Datei und dann die einzelnen Zeilen einlesen können. Die Zeilen werden zwar als String gelesen, können aber mittels `Integer.parseInt()` in `Integer`(Ganzzahlen) umgewandelt werden.

Aus der ersten Zeile wird dann die Anzahl der Spieler und somit die Größe des Spieler Arrays (`spielerListe[]`) bestimmt.

Mittels der restlichen Zeilen wird dann das Array mit Spielern befüllt, welche die Zahlen der Datei als ihren Score, also ihre Spieler-Stärke erhalten.

Jetzt können die 3 Turnierformen getestet werden. In der Methode `variantenTesten()` in der Klasse `Turnier` ist dieser Code zusammengefasst. Das Array `spielerListe` wurde davor, beim erstellen des Turniers, erzeugt.

Die Methode `variantenTesten(int wiederholungen)` geht wie folgt vor:

1. Der beste Spieler wird ermittelt.
2. Für die Anzahl der angegebenen Wiederholungen wird folgendes wiederholt:
 1. Der Sieger eines Liga-Durchlaufes wird mit dem besten Spieler verglichen.
 2. Der Sieger eines KO-Durchlaufes wird mit dem besten Spieler verglichen.
 3. Der Sieger eines KO5-Durchlaufes wird mit dem besten Spieler verglichen.

Stimmen bester Spieler und Sieger einer Turnierform überein wird der Zähler für die entsprechende Turnierform erhöht.

Die gesamten Siege und Wiederholungen werden in der Konsole ausgegeben.

3. Es wird ein Durchschnitt der Siege gebildet, indem die Anzahl der Siege durch die Wiederholungen dividiert wird.

4. Die Turnierform, welche den höchsten Durchschnitt hat, wird empfohlen.

Es werden zwei Klassen verwendet:

1. `Spieler`: Diese stellt einen Spieler mit einer Spielernummer, einem Score(Spieler-Stärke) und seinen Siegen dar.
2. `Turnier`: Hier wird eingelesen und die verschiedenen Turnier-Varianten ausprobiert.

Die Klasse `Turnier` enthält, neben anderen, die Methoden `resetSiege()`, `meistenSiege()`, `besterSpieler()`, sowie `spiele(spieler1, spieler2)` und `spiele5mal(spieler1, spieler2)`. Diese werden für die verschiedenen Turnierformen benötigt.

`besterSpieler()`: gibt den Spieler mit dem höchsten Score zurück.

`ResetSiege()` setzt alle Siege auf 0

`meistenSiege()` gibt Spieler mit den meisten Siegen zurück

`spiele(Spieler spieler1, Spieler spieler2)`: lässt zwei Spieler gegeneinander spielen, indem die Scores der beiden Spieler addiert werden und jeder Spieler einen entsprechenden Bereich erhält, je nach Spielstärke. Anschließend wird eine Zufallszahl ermittelt. Der Spieler in dessen Bereich die

Zufallszahl liegt gewinnt. Der Bereich des ersten Spielers ist von 0 bis zu seinem Score dividiert durch den addierten Score-Wert. Spieler zwei „erhält“ den restlichen Bereich. Somit hat jeder Spieler die entsprechende Chance, dass die Zufallszahl in dessen Bereich liegt, abhängig von dessen Spieler-Stärke(Score)

spiele5mal(Spieler spieler1, Spieler spieler2): lässt zwei Spieler fünf mal gegeneinander spielen und gibt den Spieler zurück , der am häufigsten gegen den anderen gewonnen hat.

Turnierformen:

Liga: Jeder Spieler spielt gegen die auf ihn folgenden Spieler. Somit kommt es zu keinen doppelten Spielen. Der Spieler mit den meisten Siegen ist Sieger des Turniers (bei Gleichstand gewinnt der Spieler mit der kleineren Spielernummer)

K.O.: Das K.O. System wird mittels Rekursion durchgeführt. Somit sind der Turnierplan, bzw. die Paare der Spielerliste folgend (1&2, 3&4, 5&6, etc.).

Die Methode die das K.O. Turnier ausführt muss immer den Anfang und das Ende der Liste erhalten, für die sie das Turnier durchführen soll.

Die Abbruchbedingung ist, wenn die Differenz von Ende und Anfang 1 ist. Hier sind noch 2 Spieler zu bearbeiten und der Gewinner der Beiden wird zurückgegeben.

Ist die Abbruchbedingung nicht erfüllt, so spielen der Gewinner der ersten Hälfte gegen den Gewinner der zweiten Hälfte. Dazu erfolgen hier zwei ein Rekursionsaufrufe.

1. spieleKOTurnier(anfang, (**int**) Math.floor((anfang + ende) / 2)))

Dieser ruft nun die Methode spieleKOTurnier, mit der ersten Hälfte der Spielerliste, auf.

2 spieleKOTurnier((**int**) Math.ceil((anfang + ende) / 2), ende)))

Dieser ruft nun die Methode spieleKOTurnier, mit der zweiten Hälfte der Spielerliste, auf.

K.O. x 5: Diese Turnierform funktioniert ähnlich, wie die K.O. Form, nur wird nicht die Methode spiele(), sondern spiele5mal() verwendet. Somit spielen die Spieler nicht nur einmal Gegeneinander, um in die nächste Runde zu gelangen, sondern fünf mal.

Die Methode spieleKO5Turnier() ist ebenfalls Rekursiv.

Beispiele

1.

Ausgewählte Datei:

8
0
10
20
30

40
50
60
100

Ausgabe in der Konsole:

Siege im Liga-System : 49998.0 bei 50000 Wiederholungen
Siege im KO-System : 17738.0 bei 50000 Wiederholungen
Siege im K05-System : 21383.0 bei 50000 Wiederholungen

Durchschnitt Liga: 0.99996
Durchschnitt K0: 0.35476
Durchschnitt K05: 0.42766
Das Liga-System wird empfohlen!

Erläuterung:

Aus der Datei wird die Größe des Arrays `spielerListe` gelesen (Zeile 1) und die restlichen Zeilen zur Befüllung des Arrays mit Spielern verwendet. Die gelesenen Zahlen sind der Score der Spieler.

Liga: Die Liga-Form funktioniert, indem der erste Spieler gegen alle darauf folgenden Spieler spielt. Danach spielt der zweite Spieler gegen alle weiteren Spieler, usw. Dies geht bis zum vorletzten Spieler. Somit hat jeder Spieler einmal gegen alle anderen gespielt.

K.O.: Die übergebenen Parameter für die Methode des K.O. Systems sind hier 0 und 7. In der Methode wird eine Unterteilung in 0 bis 3 und 4 bis 7 bestimmt. Die K.O.- Gewinner dieser Unterteilungen spielen dann gegeneinander. Die Unterteilung von 0 bis 3 erfolgt dann in 0 bis 1 und 2 bis 3. Eine weitere Unterteilung gibt es nicht mehr, da die Abbruchbedingung erfüllt ist (bzw. es noch zwei Spieler gibt). Die Unterteilung von 4 bis 7 wurde in 4 bis 5 und 6 bis 7 gemacht.

Dies sind die Paare, welche gegeneinander spielen.

Das K.O. x 5 System funktioniert nach dem selben Prinzip, nur wird die Methode `spiele5mal` verwendet. Somit spielt jedes Paar fünf mal gegeneinander.

Bewertung des Ergebnisses: Das Programm ergibt, dass der beste Spieler (hier Spieler 8) im Liga System fast immer gewinnt, die Wahrscheinlichkeit, dass er gewinnt ist im K.O. System am niedrigsten und im K.O. 5 System etwas höher ist .

Diese Ergebnisse sind sinnvoll, denn im Liga System, spielt der beste Spieler gegen alle. Er spielt also auch gegen die „schlechten“ Spieler und hat dabei die höchste Wahrscheinlichkeit zu gewinnen.

Die Wahrscheinlichkeit in den K.O. Systemen zu gewinnen ist niedriger, denn der beste Spieler ist hier der achte Spieler und er spielt zuerst gegen den siebten und dann gegen den vierten oder fünften Spieler. Er spielt somit also immer gegen die besseren Spieler. Die Wahrscheinlichkeit zu verlieren ist höher.

Das selbe gilt für das K.O. x5 System, allerdings ist die Wahrscheinlichkeit zu gewinnen höher, als beim „normalen“ K.O. System, denn über mehr Spiele ist die Wahrscheinlichkeit höher, dass der Bessere gewinnt.

2.

Ausgewählte Datei:

16
100
95
95
95
95
95
95
95
95
95
95
95
95
95
95
95
95
95

Ausgabe in der Konsole:

Siege im Liga-System : 3109.0 bei 5000 Wiederholungen
Siege im KO-System : 337.0 bei 5000 Wiederholungen
Siege im K05-System : 382.0 bei 5000 Wiederholungen

Durchschnitt Liga: 0.6218
Durchschnitt KO: 0.0674
Durchschnitt K05: 0.0764
Das Liga-System wird empfohlen!

Erläuterung:

Das vorgehen ist das selbe, wie bei der ersten Datei , nur ist die Länge der Spielerliste hier 16 (erste Zeile der Datei).

Die Turnierformen funktionieren gleich. Auch bei den K.O. Systemen wird gleichmäßig aufgeteilt, ohne das Spieler ausgelassen werden (Die übergebenen Parameter sind 0 und 15).

Da bei dieser Datei die Spielerstärken näher beieinander liegen, sind die gesamten Siege des besten Spieler deutlich niedriger als zuvor. Trotzdem ist das Liga System das empfohlene System.

Quellcode

```
public Spieler spiele(Spieler spieler1, Spieler spieler2) {  
    if (!(spieler1.getScore() == 0 && spieler2.getScore() == 0)) {  
        double x = spieler1.getScore() + spieler2.getScore();  
        double y = spieler1.getScore() / x;  
        double r = Math.random();  
        if (r < y) {  
            spieler1.erhoeheSiege();  
        }  
    }  
}
```

```

        return spieler1;
    } else {
        spieler2.erhoeheSiege();
        return spieler2;
    }
} else
    return null;
}

public Spieler spiele5mal(Spieler spieler1, Spieler spieler2) {
    double x = spieler1.getScore() + spieler2.getScore();
    double y = spieler1.getScore() / x;
    int zwischenstandSpieler1 = 0;
    int zwischenstandSpieler2 = 0;
    for (int i = 0; i < 5; i++) {
        double r = Math.random();
        if (r < y) {
            zwischenstandSpieler1++;
        } else {
            zwischenstandSpieler2++;
        }
    }
    if (zwischenstandSpieler1 < zwischenstandSpieler2) {
        spieler2.erhoeheSiege();
        return spieler2;
    } else {
        spieler1.erhoeheSiege();
        return spieler1;
    }
}

public Spieler ligaTurnier() {
    int b = 1;
    for (int i = 0; i < spielerListe.length - 1; i++) {
        for (int j = b; j < spielerListe.length; j++) {
            spiele(spielerListe[i], spielerListe[j]);
        }
        b++;
    }
    return meistenSiege();
}

public Spieler spieleKOTurnier(int anfang, int ende) {
    if (ende - anfang == 1) // Abbruchbedingung
        return spiele(spielerListe[anfang], spielerListe[ende]);
    return spiele(spieleKOTurnier(anfang, (int) (Math.floor((anfang + ende) / 2))), spieleKOTurnier((int) (Math.ceil((anfang + ende) / 2)), ende));
}

public Spieler spieleK05Turnier(int anfang, int ende) {
    if (ende - anfang == 1)
        return spiele5mal(spielerListe[anfang], spielerListe[ende]);
}

```

```
        return spiele5mal(spieleK05Turnier(anfang, (int) Math.floor((anfang +
ende) / 2)), spieleK05Turnier((int) Math.ceil((anfang + ende) / 2),
ende));
    }

    public void variantenTesten(int w) {
        Spieler besterSpieler = besterSpieler();
        int wiederholungen = w;

        double anzahlLigaSiege = 0;
        for (int i = 0; i < wiederholungen; i++) {
            if (ligaTurnier() == besterSpieler)
                anzahlLigaSiege++;
        }

        double anzahlK0Siege = 0;
        for (int i = 0; i < wiederholungen; i++) {
            if (spieleK0Turnier(0, spielerListe.length - 1) == besterSpieler)
                anzahlK0Siege++;
        }

        double anzahlK05Siege = 0;
        for (int i = 0; i < wiederholungen; i++) {
            if (spieleK05Turnier(0, spielerListe.length - 1) == besterSpieler)
                anzahlK05Siege++;
        }

        System.out.println("Siege im Liga-System : " + anzahlLigaSiege + " bei "
+ wiederholungen + " Wiederholungen");
        System.out.println("Siege im K0-System : " + anzahlK0Siege + " bei " +
wiederholungen + " Wiederholungen");
        System.out.println("Siege im K05-System : " + anzahlK05Siege + " bei " +
wiederholungen + " Wiederholungen");

        System.out.println();

        double durchschnittLigaSiege = (anzahlLigaSiege / wiederholungen);
        double durchschnittK0Siege = (anzahlK0Siege / wiederholungen);
        double durchschnittK05Siege = (anzahlK05Siege / wiederholungen);

        System.out.println("Durchschnitt Liga: " + durchschnittLigaSiege);
        System.out.println("Durchschnitt K0: " + durchschnittK0Siege);
        System.out.println("Durchschnitt K05: " + durchschnittK05Siege);

        double x = Math.max(durchschnittLigaSiege, durchschnittK0Siege);
        double max = Math.max(x, durchschnittK05Siege);

        if (max == durchschnittLigaSiege)
            System.out.println("Das Liga-System wird empfohlen!");
        if (max == durchschnittK0Siege)
            System.out.println("Das K0-System wird empfohlen!");
        if (max == durchschnittK05Siege)
            System.out.println("Das K05-System wird empfohlen!");
    }
}
```

```
}
```

```
public static void main(String[] args) {  
    Turnier t = new Turnier();  
    t.variantenTesten(5000);  
}
```