

Würfelpoker



Abbildung 1: Würfelpoker

Einführung

Würfelpoker¹ ist ein einfaches Spiel, bei dem **fünf Würfel** gewürfelt werden und die resultierende Augenzahl verwendet wird, um die Pokerhand des Spielers zu ermitteln (siehe Beschreibung von Pokerhänden unten). Jeder Würfel hat die Werte 1, 2, 3, 4, 5 und 6 auf den Seiten und der Spieler **würfelt alle fünf Würfel zu Beginn seines Spieles**. Der Spieler hat dann die Möglichkeit, **einige oder alle Würfel bis zu zwei Mal neu zu würfeln**, um zu versuchen, die bestmögliche Pokerhand zu erreichen. Nach dem dritten Wurf werden die endgültigen Würfelwerte des Spielers verwendet, um seine Pokerhand zu bestimmen, und die Hände werden entsprechend der unten erwähnten Rangfolge der Pokerhände eingestuft.

Der Spieler mit der besten Pokerhand gewinnt das Spiel. Wenn beide Spieler den gleichen Pokerhandwert haben, haben wir ein Unentschieden.

Die Poker-Würfelhände sind (sortiert nach Wert mit dem besten Pokerblatt oben):

Hand	Beschreibung	Beispiel
Five of a kind	Fünf identische Würfelwerte	3 3 3 3 3
Four of a kind	Vier identische Würfelwerte	2 2 2 2 1
Full House	Drilling und ein Paar	1 1 1 4 4
Straight	Fünf Würfelwerte in fortlaufender Rangfolge	2 3 4 5 6

¹https://en.wikipedia.org/wiki/Poker_dice

Hand	Beschreibung	Beispiel
Three of a kind	Drei identische Würfelwerte	2 2 2 5 6
Two pairs	Zwei Paare	2 2 3 3 6
One pair	Ein Paar	1 2 2 3 4
Bust	Alle Würfelwerte sind unterschiedlich	1 3 4 5 6

Levels

Diese Übung ist in vier Levels gegliedert:

- Level 1 enthält Mindestanforderungen. Mit genügend Zeit sollten alle in der Lage sein, dieses Programm ohne viel Hilfe von LehrerInnen oder anderen SchülerInnen richtig zu schreiben.
- Level 2 ist schwieriger. Sie werden wahrscheinlich ein wenig Hilfe von LehrerInnen oder fortgeschrittenen SchülerInnen benötigen, um es fertigzustellen.
- Level 3 ist für fortgeschrittene SchülerInnen, die ihre Programmierkenntnisse noch weiter verbessern möchten.
- Level 4 bringt alles zusammen. Sobald Sie Level 3 gelöst haben, sollte Level 4 kein großes Problem sein.

Level 1: Mindestanforderungen

In diesem Level müssen Sie ein Einzelspieler-Spiel ohne Auswertung der Pokerhand umsetzen.

Zunächst ein Tipp zum Einstieg: Erstellen Sie fünf Variablen für die fünf Würfel Ihres Spiels. Erstellen Sie außerdem fünf Variablen, die die Information speichern, welche Würfel fixiert sind (d.h. in der nächsten Runde nicht neu gewürfelt werden).

```
int würfel1 = 0, würfel2 = 0, würfel3 = 0, würfel4 = 0, würfel5 = 0;
bool Fixed1 = falsch, Fixed2 = falsch, Fixed3 = falsch, Fixed4 = falsch,
↪ Fixed5 = falsch;
```

- Implementieren Sie eine Methode `RollDice`. Sie soll allen Würfeln, die derzeit nicht fixiert sind, eine Zufallszahl zwischen 1 und 6 zuweisen.
- Implementieren Sie eine Methode `PrintDice`. Sie soll die Werte aller Würfel auf dem Bildschirm ausgeben (zB 2, 2, 1, 6, 5).
- Implementieren Sie eine Methode `FixDice`.

- Zuerst werden alle vorherigen Würfel fixierungen zurückgesetzt. Alle Würfel würden neu gewürfelt.
 - Der Benutzer kann Würfel fixieren/lösen, indem er die Würfelnummer eingibt (z. B. die Eingabe von 1 fixiert den ersten Würfel, wenn dieser zuvor gelöst war; andernfalls löst es den ersten Würfel). Sie können `Console.ReadLine` oder `Console.ReadKey` verwenden, was immer Sie bevorzugen.
 - Fragen Sie den/die BenutzerIn so lange nach fixierten Würfeln, bis er/sie `r` drückt (für "roll the dice").
- Implementieren Sie die Hauptspiellogik mit den oben genannten Methoden.
 1. Würfle (Methode `RollDice`) und schreibe die Würfelwerte auf dem Bildschirm (Methode `PrintDice`).
 2. Fragen Sie den/die BenutzerIn nach fixierten Würfeln (Methode `FixDice`). Falls *alle* Würfel fixiert sind, endet das Spiel.
 3. Wiederholen Sie die Schritte 1 und 2.
 4. Würfle zum letzten Mal und gib die Würfelwerte auf dem Bildschirm aus.

Dies ist eine beispielhafte Ein-/Ausgabe für Ebene 1:

```
Dice roll #1 (out of 3): 2, 2, 1, 6, 5
Which dice do you want to fix/unfix? (1-5, or 'r' to roll the dice)
1
Fixed: 1
2
Fixed: 1 2
3
Fixed: 1 2 3
3
Fixed: 1 2
r
---
Dice roll #2 (out of 3): 2, 2, 5, 5, 4
Which dice do you want to fix/unfix? (1-5, or 'r' to roll the dice)
1
Fixed: 1
2
Fixed: 1 2
3
Fixed: 1 2 3
4
Fixed: 1 2 3 4
r
```

Dice roll #3 (out of 3): 2, 2, 5, 5, 3

Level 2: Würfel sortieren

Schreiben Sie eine Methode `SortDice`, die die Würfel nach Wert sortiert. `dice1` sollte den niedrigsten Wert enthalten, `dice5` den höchsten Wert. Hier sind einige Beispiele:

Vor Sortierung	Nach Sortierung
4 3 2 6 5	2 3 4 5 6
6 3 1 6 6	1 3 6 6 6
5 5 1 2 1	1 1 2 5 5

Fügen Sie die `SortDice`-Methode zur Hauptspiellogik hinzu. Rufen Sie sie ganz am Ende auf, um die Würfel nach Wert zu sortieren und ihre Werte mit der Methode `PrintDice` auf dem Bildschirm auszugeben.

Zum Sortieren können Sie den Algorithmus *Bubble Sort* verwenden. Er ist einfach. Vertauschen Sie einfach den Wert zweier benachbarter Würfel, wenn sie nicht in der richtigen Reihenfolge sind. Tun Sie dies, bis alle Würfel in der richtigen Reihenfolge sind. Das Video *Bubble Sort with Hungarian, Folk Dance*² demonstriert das Prinzip (das Sortieren selbst beginnt ab hier³).

Hier ist ein Beispiel:

- Schauen wir uns die Startwürfelwerte 4 3 2 6 5 an.
- Erste Runde:
 - Wir beginnen mit dem ersten Paar 4 und 3. 4 ist größer als 3, also vertausche die Werte. Wir haben jetzt 3 4 2 6 5.
 - Das nächste Paar ist jetzt 4 und 2. 4 ist größer als 2, also vertausche die Werte. Wir haben jetzt 3 2 4 6 5.
 - Das nächste Paar ist jetzt 4 und 6. 4 ist nicht größer als 6, also vertauschen Sie die Werte nicht. Wir haben immer noch 3 2 4 6 5.
 - Das letzte Paar ist 6 und 5. 6 ist größer als 5, also vertausche die Werte. Wir haben jetzt 3 2 4 5 6.
 - Sind die Würfel schon sortiert? Nein, also wiederhole alles von vorne.

²<https://youtu.be/lv3vgjM8Pv4>

³<https://youtu.be/lv3vgjM8Pv4?t=53>

- Zweite Runde:
 - Wir beginnen mit dem ersten Paar 3 und 2. 3 ist größer als 2, also vertausche die Werte. Wir haben jetzt 2 3 4 5 6.
 - Alle anderen Paare sind bereits sortiert, sodass keine weiteren Vertauschungen stattfinden.
 - Sind die Würfel jetzt sortiert? Ja, wir sind fertig.
- Das Ergebnis ist 2 3 4 5 6.

Level 3: Ergebnis analysieren

Schreiben Sie eine Methode `AnalyzeAndPrintResult`. Es wird davon ausgegangen, dass die Variablen `dice1` bis `dice5` Werte in aufsteigender Reihenfolge enthalten. Die Methode ermittelt aus den Würfeln die Pokerhand (zB *Full House*) und gibt sie auf dem Bildschirm aus.

Fügen Sie der Hauptspiellogik die Methode `AnalyzeAndPrintResult` hinzu. Rufen Sie sie ganz am Ende auf, um die Pokerhand des/der SpielerIn auszugeben.

Level 4: Zwei-Spieler-Modus

- Extrahieren Sie die Spiellogik in eine `Play`-Methode und führen Sie sie zweimal aus, einmal für jede/n SpielerIn.
- Schreiben Sie eine Methode `DetermineWinner`, die die Pokerhände beider Spieler vergleicht und ausgibt, ob wir ein Unentschieden haben. Wenn nicht, wird ausgegeben, wer der/die GewinnerIn ist.

Dies ist eine beispielhafte Ein-/Ausgabe für Level 4:

```
PLAYER 1:
=====
Dice roll #1 (out of 3): 2, 2, 1, 6, 5
Which dice do you want to fix/unfix? (1-5, or 'r' to roll the dice)
1
Fixed: 1
2
Fixed: 1 2
3
Fixed: 1 2 3
3
Fixed: 1 2
r
---
```

Dice roll #2 (out of 3): 2, 2, 5, 5, 4

Which dice do you want to fix/unfix? (1-5, or 'r' to roll the dice)

1

Fixed: 1

2

Fixed: 1 2

3

Fixed: 1 2 3

4

Fixed: 1 2 3 4

r

Dice roll #3 (out of 3): 2, 2, 5, 5, 3

Two pairs!

PLAYER 2:

=====

Dice roll #1 (out of 3): 2, 2, 3, 5, 4

Which dice do you want to fix/unfix? (1-5, or 'r' to roll the dice)

2

Fixed: 2

3

Fixed: 2 3

4

Fixed: 2 3 4

5

Fixed: 2 3 4 5

r

Dice roll #2 (out of 3): 5, 2, 3, 5, 4

Which dice do you want to fix/unfix? (1-5, or 'r' to roll the dice)

2

Fixed: 2

3

Fixed: 2 3

4

Fixed: 2 3 4

5

Fixed: 2 3 4 5

r

Dice roll #3 (out of 3): 2, 2, 3, 5, 4

One pair!

Player 1 wins!