

RollADice

Ein unterhaltsames Spiel aus der guten, alten DOS-Konsolen-Zeit funktioniert folgendermaßen:

Ein Würfel wird zwölfmal geworfen. Kommt eine Sechs, kann der/die SpielerIn sie behalten (falls er/sie noch keine genommen hat). Falls der/die SpielerIn die letzte Sechs im Spiel gewählt hat, ist er SiegerIn, ansonsten gewinnt die Bank. Falls gar keine Sechs gewürfelt wird, gewinnt ebenfalls der/die SpielerIn.

Verwenden Sie einen Zufallsgenerator , um den Würfel zu simulieren. Ein(e) SpielerIn kann nur eine Sechs behalten. Daher werden, sobald der/die SpielerIn eine Sechs behält, alle weiteren Züge ohne BenutzerInneninteraktion ausgeführt.

Verwenden Sie Variablen, um den Programmfluss zu steuern. Der Rest ist eine Schleife und ein paar ifs. Im Folgenden gibt es je ein Beispiel für eine gewonnene und eine verlorene Runde:

Das Spiel beginnt...

Wurf: 6 (Noch 11 Runden)

behalten? (0=Nein/1=Ja)

\stdin{0}

Wurf: 2 (Noch 10 Runden)

Wurf: 2 (Noch 9 Runden)

Wurf: 6 (Noch 8 Runden)

behalten? (0=Nein/1=Ja)

\stdin{0}

Wurf: 4 (Noch 7 Runden)

Wurf: 3 (Noch 6 Runden)

Wurf: 5 (Noch 5 Runden)

Wurf: 1 (Noch 4 Runden)

Wurf: 5 (Noch 3 Runden)

Wurf: 5 (Noch 2 Runden)

Wurf: 4 (Noch 1 Runden)

Wurf: 5 (Noch 0 Runden)

Sie haben verloren! Es wurde keine weitere Sechs gewürfelt.

Das Spiel wird beendet...

Das Spiel beginnt...

Wurf: 5 (Noch 11 Runden)

Wurf: 4 (Noch 10 Runden)

Wurf: 2 (Noch 9 Runden)

Wurf: 6 (Noch 8 Runden)

behalten? (0=Nein/1=Ja)

\stdin{0}

Wurf: 4 (Noch 7 Runden)

Wurf: 3 (Noch 6 Runden)

Wurf: 1 (Noch 5 Runden)

Wurf: 3 (Noch 4 Runden)

Wurf: 6 (Noch 3 Runden)

```
behalten? (0=Nein/1=Ja)
\stdin{1}
Restliche Würfe: 5 3 4
Sie haben gewonnen!
Das Spiel wird beendet...
```