

Programmbeschreibung und Flussdiagramm

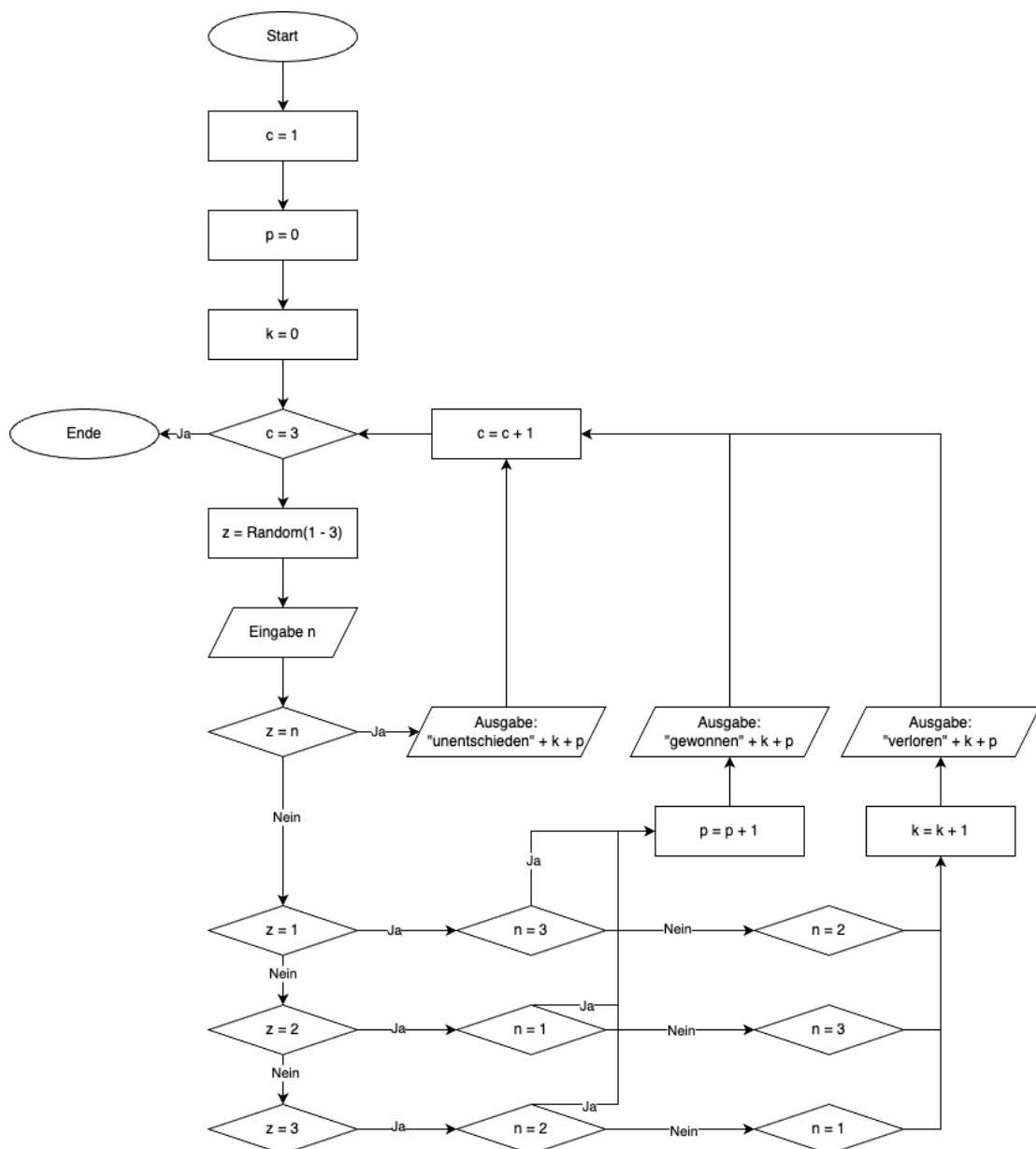


Hinweis

Dieses Programm spielt mit dem User „Stein, Schere, Papier“. Dabei werden Punkte vergeben.

Die Umsetzung des Programm verwendet:

- while-Schleife
- if-Abfrage
- Variablen
- Inkrementieren



Schritt-für-Schritt-Anleitung

In dem folgenden Teil führt „Schritt-für-Schritt“ durch das Flussdiagramm.

Am Ende dieser Anleitung steht das vollständige Programm passend zum Flussdiagramm.

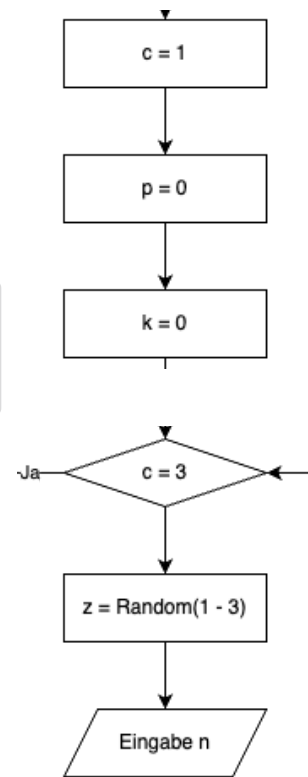
1. Schritt: Die Eingabe (Zeile 1 und 3)

„let“ deklariert die Variablen z, g und c.

Die Variable **c** dient zum „Hochzählen“, um die Anzahl der Versuche einzuhalten. **p** zählt die Punkte vom User und **k** die Punkte vom „Computer“.

JavaScript

```
1 let c = 1;
2 let p = 0;
3 let k = 0;
```



2. Schritt: Die while-Schleife (Zeile 5 bis 7)

Solange **c kleiner 4** (also 3 Versuche) ist, läuft das Spiel durch.

Die Variable **n** wird mit dem Wert aus "prompt" deklariert. Der Zähler z erzeugt Zufallszahlen: 1 für Stein, 2 für Schere und 3 für Stein.

JavaScript

```
1 let c = 1;
2 let p = 0;
3 let k = 0;
4
5 while (c < 4) {
6   let z = Math.floor(Math.random() * 3) + 1;
7   let n = prompt('Wähle Stein (1), Schere (2), Papier (3): ');
```

3. Schritt: Die Abzweigungen und Funktionen (Zeile 9 bis 41)

Da sich die Aussagen wiederholen zu „Gewonnen“ und „Verloren“ gibt es zwei Funktionen „getVerloren()“ und „getGewonnen()“. Die Abfrage unterscheidet zunächst, ob ein „Unentschieden“ gegeben ist. Danach erfolgt die Fallunterscheidung zwischen „Gewonnen“ und „Verloren“.

JavaScript

```
1 let c = 1;
2 let p = 0;
3 let k = 0;
4
5 while (c < 4) {
6   let z = Math.floor(Math.random() * 3) + 1;
7   let n = prompt('Wähle Stein (1), Schere (2), Papier (3): ');
8
9   if (z == n) {
10    alert('Unentschieden ' + p + "/" + k);
11
12   } else if (z == 1) {
13     if (n == 2) {
14       getVerloren();
15     } else {
16       getGewonnen();
17     }
18   } else if (z == 2) {
19     if (n == 3) {
20       getVerloren();
21     } else {
22       getGewonnen();
23     }
24   } else if (z == 3) {
25     if (n == 1) {
26       getVerloren();
27     } else {
28       getGewonnen();
29     }
30   }
31   c++;
32 }
33 function getGewonnen() {
34   p += 1;
35   alert('Gewonnen ' + p + "/" + k);
36 }
37
38 function getVerloren() {
39   k += 1;
40   alert('Verloren ' + p + "/" + k);
41 }
```

ENDE

Aufgaben

- ① Füge in dem Quellcode passend zu den Schritten Kommentare ein, die den jeweiligen Teil in seiner Funktion beschreiben.

JavaScript

```

1 // Zwei "Balken" ergeben einen Zeilenkommentar
2
3 /* Balken und Sternchen am Anfang und am Ende
4 ergeben einen Kommentar auf mehreren Zeilen */

```

- ② Überlege Dir mögliche Verbesserungen für das Programm. Beispielsweise:
- Könnte der Text der Eingabe und der Ausgabe verbessert werden.
 - Lässt sich der Schwierigkeitsgrad erhöhen?
 - Mit „logischen Operatoren“ wie && (Und-Verknüpfung) und || (Oder-Verknüpfung) könnte sich der Quellcode vereinfachen lassen, oder?

- ③ Passe das Flussdiagramm und den Quelltext entsprechend Deiner Überlegungen aus Aufgabe 2 an.