

# **Semesterarbeit 2023**

**von Hanna Bühler, Vlada Svinova und Jule Ziegler**



**Spiel: Held gegen Schurke**  
**Programmieren - Alexander Auch**

## **Organisatorische Daten**

Gruppenmitglieder:	Hanna Bühler	Vlada Svinova	Jule Ziegler
Matrikelnummern:	1413906	9680407	2289705

Aktivitäten:

Alle Gruppenmitglieder haben im gleichen Umfang am Projekt mitgewirkt.

Es wurde mehrere Tage zusammen am Spiel programmiert und an der Dokumentation gearbeitet. Die einzelnen Aktivitäten und Schritte wurden daher auch kaum aufgeteilt.

Am Anfang wurden einige Spielideen gesammelt, von welchen die Beste am Ende ausgewählt wurde. Durch unser begrenztes Wissen und der fehlenden Kenntnisse, bzw. Erfahrungen in Java konnten mehrere Ideen wie z. B. »Jump and Run Spiele" nicht umgesetzt werden.

Die durchgeführten Aktivitäten bestanden hauptsächlich aus der Ideenfindung, der eigentlichen Programmierung der einzelnen Klassen und der kontinuierlichen Verbesserung unseres Codes.

## **Projektidee - Was ist das Spielprinzip?**

Der Spieler des Spiels kann sich am Anfang als ersten Schritt einen Heldenamen vergeben, unter dem er spielt.

Danach kann er selbst einen aus drei verfügbaren Heldentypen auswählen, mit welchem er spielen möchte.

Jeder Heldentyp hat dabei unterschiedliche Werte in seiner Lebensanzahl, in seiner Abwehr und im Angriff. Der erste Heldentyp hat dabei einen Wert von 50 in der Lebensanzahl, der Zweite dagegen einen Wert von 75 und der Dritte 40. In der Abwehr hat der Erste einen Wert von 10, der zweite Heldentyp einen Wert von 5 und der Dritte 15. Im Angriff hat der erste Heldentyp 25, der Zweite 20 und der Dritte einen Wert von 35.

Der Spieler wählt sich im darauffolgenden Schritt einen Gegenspieler, in unserem Fall einen Schurken, aus. Die Anzahl der verfügbaren Schurken beträgt auch wie bei den unterschiedlichen Heldentypen auch 3, diese haben je unterschiedliche Lebenspunkte, Abwehrrpunkte und Angriffspunkte.

Der Kampf beginnt nach der Auswahl beider Typen.

Nun kann der Spieler selbst als Held gegen den Schurken kämpfen und immer auswählen, welche Aktion er im nächsten Schritt ausführen möchte.

Wenn er a wählt, wird ein Angriff seinerseits ausgeführt und dem Schurken werden Lebenspunkte abgezogen.

Wenn b gewählt wird, wird der Schaden, den der Schurke dem Held durch einen Angriff zufügt, durch einen Angriffsblock vermindert. Trotzdem werden dem Held einige Lebenspunkte abgezogen.

Das Spiel wird so lange gespielt, bis die Lebenspunkte des Schurken aufgebraucht sind. Es handelt sich bei dem Held gegen Schurke Spiel, das von uns programmiert wurde, um objektorientiertes Programmieren.

## Code - Struktur

Es wurden zu Beginn drei Klassen erstellt: Main, Held und Schurke.

Die Main-Klasse beschreibt dabei den Spielablauf des Kampfes, die Held- und die Schurken Klasse beinhalten jeweils die einzelnen Angaben zum Namen und den Typen des Helden und des Schurken.

Hier wird auch durch die Initialisierung festgelegt welcher Helden - und Schurkentyp wie viele Lebenspunkte, Angriffspunkte und Abwehrrpunkte hat. Zuvor wurden diese einzelnen Variablen deklariert.

Die Helden- und die Schurkenmethoden werden jeweils durch die Initialisierung und Deklaration in die Main-Klasse importiert, sodass diese Zugriff auf die anderen zwei Klassen hat und durch die verschiedenen Variablen der Kampf durchgeführt werden kann.

Um die einzelnen Variablen für die Lebenspunkte, Angriffspunkte und Abwehrrpunkte, die in die Main-Klasse importiert wurden, nicht jedes Mal ganz ausschreiben zu müssen (wie z.B. held.hph), wurden diese jeweils noch kürzer definiert und initialisiert (hph).

Der offizielle Kampf beginnt und wird mit einer While-Schleife weitergeführt, die sich solange wiederholt bis die Leben des Schurken aufgebraucht sind, also einen Wert von 0 haben.

In dieser Schleife wird der Spieler durch eine Scanner-Klasse in der Java Konsole dazu aufgefordert, eine Aktion auszuführen.

Danach wird eine switch-case-Verzweigung in die While-Schleife eingebunden.

Wenn a (case 1) eingegeben wird, führt der Held einen Angriff gegen den Schurken aus. Dabei wird der Schaden berechnet und von den Lebenspunkten des Schurken abgezogen.

Wenn dagegen b (case 2) eingegeben wird, blockt der Held den Angriff des Schurken ab. Hier wird der Schaden, den der Schurke anrichtet, durch einen Block vermindert und ein geringer Wert (deswegen + 5) von den Lebenspunkten des Helden abgezogen.

Wenn die Schurkenleben den Wert kleiner gleich Null haben, wird das Kampfspiel automatisch beendet und es kann keine weitere Aktion ausgeführt werden.

In der Java Konsole wird der Satz "Das Spiel ist vorbei! Vielen Dank Held [eingegebener Heldenname], du hast den Schurken besiegt!" ausgegeben.

### Variablenbezeichnungen:

hph = Lebenspunkte des Helden

angriff = Angriffspunkte des Helden

abwehr= Abwehrrpunkte des Helden

hps= Lebenspunkte des Schurken

sangriff=Angriffspunkte des Schurken

armor= Abwehrrpunkte des Schurken

i= Zählvariable, wird benutzt, dass die Endmeldung nur einmal angezeigt wird

## Fehler während des Programmierens

Zu Beginn traten einige Schwierigkeiten innerhalb des Programms auf.

Diese beinhalteten unter anderem einen Fehler in der Implementierung der Held- und Schurkenmethode in die Main-Klasse aufgrund von fehlenden Erfahrungen mit Java.

Es gab auch Fehler in der while-Schleife innerhalb der Main-Methode. Der System.out.println-Befehl wurde dabei erst nach der Scanner-Klasse platziert. Als Folge wurde der Spieler deshalb nicht zur Aktion aufgefordert.

Außerdem traten Probleme mit den switch-case-Befehlen auf, da wir es nicht hinbekommen haben, dass bei der Ausgabe von einem case, jeweils die Lebenspunkte des Schurken und des Helden beide berechnet ausgegeben werden.

Ein anderer Fehler, der auftrat, war die Ausgabe "Das Spiel ist vorbei! Vielen Dank Held [eingegebener Heldenname], du hast den Schurken besiegt!", die uns anfangs unendlich oft nach dem Spielende auf der Konsole ausgegeben wurde.

## Fehlerlösung

Die Implementierung der Held- und Schurkenmethode in die Main-Klasse wurde durch die folgenden Befehle gelöst:

```
Held held = new Held();
int hph = held.hph;
int angriff = held.angriff;
int abwehr = held.abwehr;

Schurke schurke = new Schurke();
int hps = schurke.hps;
int sangriff = schurke.sangriff;
int armor = schurke.armor;
```

Um das Problem in der while-Schleife innerhalb der Main-Methode zu lösen, wurden der System.out.println-Befehl und die Scanner-Klasse einfach vertauscht. Dadurch wird der Spieler direkt zur Aktion aufgefordert, nachdem der Kampf beginnt.

Der Fehler in dem switch-case-Befehl wurde gelöst, indem ein break-Befehl nach dem case 1 eingefügt wurde und beide System.out.println-Befehle für die Lebenspunkte des Schurken und des Helden für den case 2 nochmals implementiert wurden.

Für unsere Ausgabe wurde eine Zählervariable i eingefügt, dass wenn die Lebenspunkte des Schurken kleiner gleich Null sind, der Ausgabesatz "Das Spiel ist vorbei! Danke Held... " nur einmal erscheint (i < 1, i = i + 1).

## Codes:

## Main:

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);
        System.out.println("Heldnamen eingeben:");
        String heldn = scanner.next();
        System.out.println("Heldname: " + heldn);
        System.out.println("");

        Held held = new Held();
        int hph = held.hph;
        int angriff = held.angriff;
        int abwehr = held.abwehr;

        Schurke schurke = new Schurke();
        int hps = schurke.hps;
        int sangriff = schurke.sangriff;
        int armor = schurke.armor;

        int i = 0;
        int heldschaden = angriff - armor;
        int schurkeschaden = sangriff - abwehr;

        System.out.println("Der Kampf beginnt!");

        while (hph > 0) {
            if (hps <= 0) {
                if (i < 1) {
                    System.out.println("Das Spiel ist vorbei! \nVielen  
Dank Held " + heldn + ", du hast den Schurken besiegt!");
                    i = i + 1;
                }
            }

            while (hps > 0) {
                System.out.println("Aktion eingeben, a für Angriff, b für  
Blockangriff:");
                System.out.println("");

                Scanner scanner2 = new Scanner(System.in);
                String aktion = scanner2.next();

                switch (aktion) {
                    case "a":
                        System.out.println("ANGRIFF!");
                        hps = hps - heldschaden;
                        System.out.println("Schurkenlebenspunkte: " + hps);
                        hph = hph - schurkeschaden;
                        System.out.println("Heldenlebenspunkte: " + hph);
                        System.out.println("");
                        break;

                    case "b":
                        System.out.println("BLOCKANGRIFF!");
                        hps = hps - heldschaden;
                        System.out.println("Schurkenlebenspunkte: " + hps);
                        hph = hph - schurkeschaden + 5;
                        System.out.println("Heldenlebenspunkte: " + hph);
                        System.out.println("");
                        break;
                }
            }
        }
    }
}
```

## Held und Schurke:

```
import java.util.Scanner;

public class Held {
    int hph;
    int abwehr;
    int angriff;

    public Held() {
        System.out.println("Helddtyp
        eingeben: ");

        Scanner scanner = new
        Scanner(System.in);

        int typ = scanner.nextInt();

        System.out.println("Helddtyp: " +
        typ);
        System.out.println(" ");
    }
}
```

```
        switch (typ) {
            case 1:
                hph = 50;
                abwehr = 10;
                angriff = 25;
                break;

            case 2:
                hph = 75;
                abwehr = 5;
                angriff = 20;
                break;

            case 3:
                hph = 40;
                abwehr = 15;
                angriff = 35;
                break;
        }
```

```
    }
}
```

```
import java.util.Scanner;

public class Schurke {
    int hps;
    int armor;
    int sangriff;

    public Schurke() {
        System.out.println("Schurkentyp
        eingeben: ");

        Scanner scanner = new
        Scanner(System.in);

        int styp = scanner.nextInt();

        System.out.println("Helddtyp: " + typ);
        System.out.println(" ");
    }
}
```

```
        switch (styp) {
            case 1:
                hps = 60;
                armor = 5;
                sangriff = 10;
                break;

            case 2:
                hps = 30;
                armor = 2;
                sangriff = 15;
                break;

            case 3:
                hps = 45;
                armor = 3;
                sangriff = 12;
                break;
        }
```

```
    }
}
```