GTA Einführung Robotik mit Makecode

Mattias Schlenker

Wilhelm-Ostwald-Gymnasium

7. Januar 2021

Wir verbessern den Space-Shooter

Viele von uns haben am Shooter weiter programmiert, daher gab es einige Fragen zu Spielmechanik, aber auch Fehlersuche und "Best Practices" hinsichtlich Übersichtlichkeit. Wichtig: Wir haben uns die Funktion der Kollisionserkennung noch einmal im Detail angesehen.

Pizza!

Neu eingeführt haben wir Versorgungsraumschiffe, die wir der Bequemlichkeit halber mit einem Pizza-Sprite versehen haben und die etwa alle fünf Sekunden erscheinen sollen. Maßgabe war, dass fünf Pizza-Sprites ein neues Leben ergeben. Für die Pizza-Sprites haben wir eine neue Kategorie "Food" eingeführt:

```
venn Spielupdate alle 1000 ▼ ns

setze projectile ▼ auf Projektil 🍄 von der Seite mit vx ① vy vGegner ▼

setze projectile ▼ Position auf x wähle eine zufällige Zahl von ② bis 160 y ②

setzte projectile ▼ Art auf Enemy ▼

wenn ② % Chance dann

setze foodProjectile ▼ auf Projektil ❷ von der Seite mit vx ② vy ⑥

setzte foodProjectile ▼ Position auf x wähle eine zufällige Zahl von ② bis 160 y ②

setzte foodProjectile ▼ Art auf Food ▼
```

Figure: Pizza!

Protokollierung

Viele Entwicklungsumgebungen bieten die Möglichkeit, Werte oder Text zu protokollieren. Das nennt sich "Debugging-Ausgaben schreiben". Makecode bietet dafür die Klasse "Konsole". Nutzt sie! Initialisiert eine Variable und schreibt sie nach jeder Änderung auf die Konsole.

```
bein Start

setze Punktzahl auf 0

setze vGegner → auf 60

Startbildschirn Sternenfeld → Effekt ⊕

setze nySprite → auf Sprite → der Art Player →

bewege nySprite → nit Knöpfen ⊕

setze nySprite → in Bildschirn bletben → the setze foodRations → auf 0

Konsole protokolliere foodRations →
```

Figure: Konsolen-Logging

Woher kommt das Wort "Debugging"

Ein "Bug" ist ein Käfer, aber was hat der in einem Programm zu suchen? Als die Logikgatter noch Relais (elektromagnetische Schalter) waren, sagten Programmierer gerne: "Da ist sicher ein Insekt im Schalter. ICH habe richtig programmiert." 1947 wurde tatsächlich mal eine Motte in einem Relais gefunden:

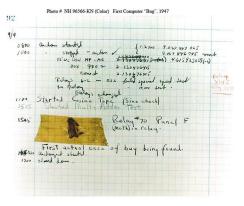


Figure: Debugging

Konsolenausgabe

Ihr schaltet die Konsolenausgabe im laufenden Spiel über den Button "Konsole anzeigen" an.

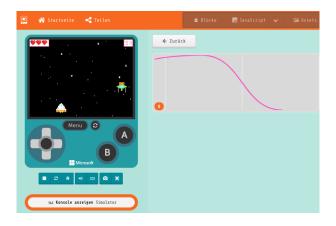


Figure: Debug-Ausgabe

Variablen als Referenzen übergeben

Wenn ein Sprite mit einem anderen kollidiert, kennen wir deren Namen nicht. Daher wird eine Referenz erzeugt und diese in die Funktion hinein übergeben. Verwendet innerhalb der Funktion nur die Referenzen "sprite" und "otherSprite"!

```
wenn sprite der Art Player ▼ überlappt otherSprite der Art Enemy ▼

zerstöre mySpr otherSprite
```

Figure: Kollisionserkennung

Best Practices

Wie mit einer Vielfalt von Gegnern und Essensrationen umgehen, ohne das Spiel unübersichtlich zu machen?

Nutzt möglichst wenige Blöcke des Typs "Spielupdate".

Verwendet stattdessen innerhalb eines Blockes Verzweigungen, die Zufallsvariablen nutzen, um verschiedene Gegner zu spawnen oder gelegentlich mal eine Pizza zu droppen.

Spiele weitergeben

Klickt auf das Diskettensymbol, um das Spiel zu speichern. Die Bild-Datei im PNG-Format enthält einen Screenshot und den Programmcode. Natürlich könnt Ihr auch "Herunterladen" klicken, das erzeugt fertige UF2-Dateien für Spielekonsolen. Und mit "Teilen" erzeugt Ihr einen Link für Mitschüler und mich.



Figure: Herunterladen