

1. Einführung einer Schulklasse in das Frogger Spiel

Bevor wir mit dem Programmieren und Gestalten loslegen, sollte den Schülerinnen und Schülern das Frogger-Spiel geläufig sein und einige Vorüberlegungen gemacht werden.

Dazu kann man das Frogger-Szenario mit jüngeren Kindern nachspielen, oder man schaut sich auf Youtube ein altes Atari Frogger-Spiel an. Fragen Sie die Schüler folgendes:

- Was muss der Frosch machen?
- Wie kann man das Spiel gewinnen?
- Wann verliert der Spieler?
- Wie kommt der Frosch heil über die Strasse?
- Was passiert, wenn der Frosch überfahren wird?
- Kann der Frosch schwimmen?
- Was machen die Krokodile?
- Warum blinken die Baumstämme manchmal und was passiert, wenn ein Frosch auf einem Baumstamm aus dem Spielfeld heraustreibt?
- Welche Besonderheiten hat das Spiel?

Nun kann die Lehrperson die Schulklasse darauf hinführen, wie es gelingt, dass der Computer dieses Spiel so interpretiert, dass es dem Spielverlauf entspricht.

Dazu kann man den Text z.B. vorlesen, aushändigen oder an die Tafel projizieren:

Du bist ein Frosch. Deine Aufgabe ist einfach: hüpf über eine befahrene Straße, weiche Fahrzeugen aus, bis du ans Ufer eines Flusses gelangst, wo du ohne zu ertrinken zur Grotte ans andere Ufer gelangen sollst, indem du auf den Rücken von Schildkröten und Baumstämme springst. Hüte dich vor Schlangen und Krokodilen! (Sega, 1980)

Zuerst sollen die Schülerinnen und Schüler die Objekte identifizieren und benennen:

Du bist ein **Frosch**. Deine Aufgabe ist einfach: hüpf über eine befahrene **Straße**, weiche **Fahrzeugen** aus, bis du ans **Ufer** eines **Flusses** gelangst, wo du ohne zu ertrinken zur **Grotte** ans andere **Ufer** gelangen sollst, indem du auf den Rücken von **Schildkröten** und **Baumstämme** springst. Hüte dich vor **Schlangen** und **Krokodilen** !

Nun erklären Sie der Klasse, dass es Bewegungen im Spiel gibt, welche vom Spieler initiiert werden und solche, welche von den Objekten selbstständig ausgeführt werden. Man unterscheidet beim Programmieren folgende Objektsteuerungen:

1. Benutzer gesteuert (user controlled)
2. Objekte mit künstlicher Intelligenz (artificial intelligence)
3. Oft hat man eine Mischung aus beidem

Fragen Sie die Schülerinnen und Schüler:

- Welche Möglichkeiten gibt es im Allgemeinen, dass der Spieler ein Objekt steuert? (Tastatur, Spracheingabe, Joystick - in unserem Fall beschränkt sich die Eingabemöglichkeit auf die Tastatur).
- Was könnte man mit künstlicher Intelligenz bewirken? (z.B. gleichmässige Bewegungen, zufällige Bewegungen, zielgerichtete Bewegungen, das Beenden des Spieles beim Aufprall des Frosches...)
- In welchem Objekt vermischen sich die beiden Steuermechanismen (beim Frosch)
- Welche Objekte sind im Spiel vom Benutzer gesteuert und welche Objekte haben künstliche Intelligenz?

Erkennen von Bewegungsmustern: Dazu können Sie wieder die Spielbeschreibung hinzuziehen und die Schülerinnen und Schüler die Verben identifizieren lassen:

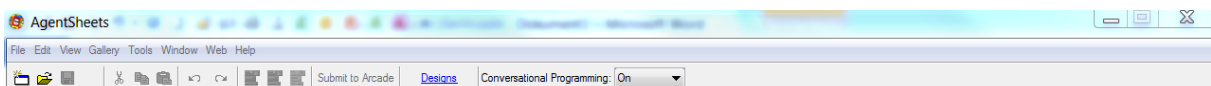
Du bist ein Frosch. Deine Aufgabe ist einfach: **hüpfe** über eine befahrene Straße, **weiche** Fahrzeugen **aus**, bis du ans Ufer eines Flusses gelangst, wo du ohne zu **ertrinken** zur Grotte ans andere Ufer **gelangen** sollst, indem du auf den Rücken von Schildkröten und Baumstämme **spingst**.

Weisen Sie die Klasse darauf hin, dass der Computer diese Begriffe nicht versteht. Diese müssen später sozusagen übersetzt werden bzw. es gilt, diese Begriffe (Verben) den Programmdialogen zuzuordnen.

An dieser Stelle können die Schülerinnen und Schüler noch überlegen, welche Interaktionen zwischen den Objekten stattfinden.

- Fahrzeuge fahren entlang einer Strasse.
- Baumstämme und Krokodile bewegen sich auf einem Fluss.
- Fahrzeug überfährt den Frosch, wenn dieser nicht ausweicht.
- Frosch hüpft auf Baumstamm oder Schildkröte, um von dieser transportiert zu werden.

Nun können die Lernenden ans System und die Software starten. Im folgenden benutzen wir die 2D Software AgentSheets (www.agentsheets.com). Nach dem Öffnen der Software erscheint im oberen Bereich des Bildschirms ein Menü:

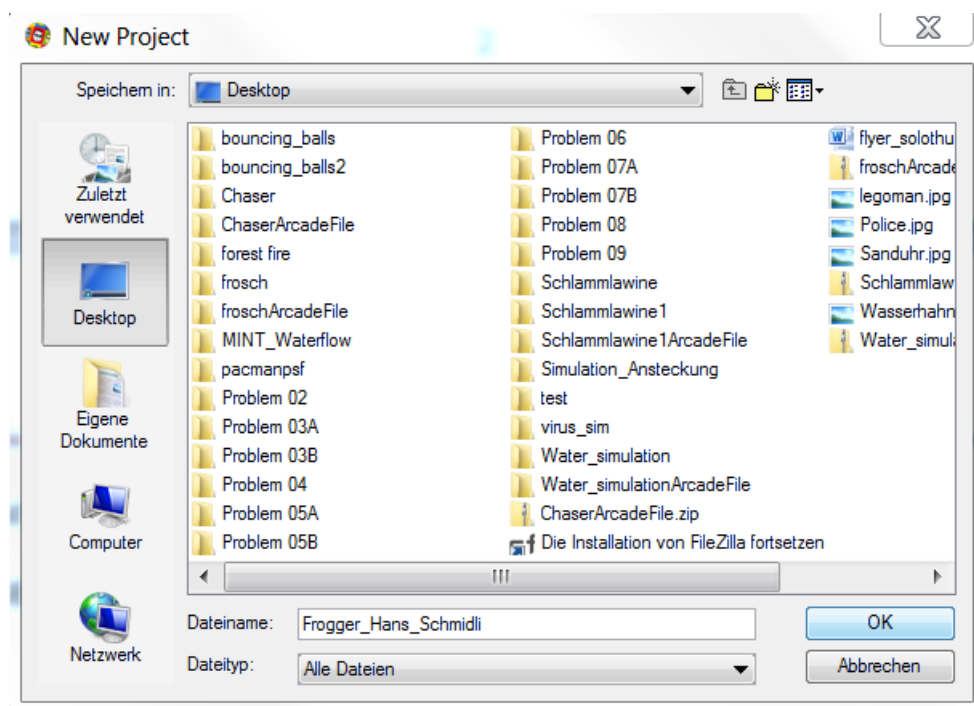


Hier wählen die Schülerinnen und Schüler folgende Menü-Navigation:

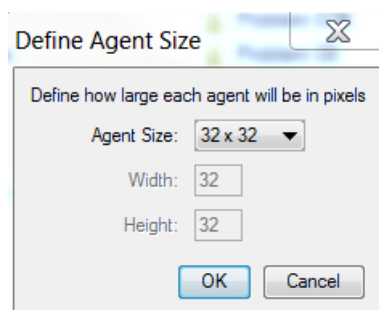
FILE -> NEW PROJECT

Die Namensgebung für das Projekt sollte so sein, dass der Lehrer die Programme später eventuell auf einem Server hochladen kann und anhand der Namen zuordnen kann. Z.B. Könnte der Name wie folgt lauten: Frogger_Hans_Schmidli

Gespeichert wird das Programm-File entweder auf dem Desktop oder unter einem beliebigen Dateipfad.



Nun erscheint folgendes Dialogfenster, welches die Schülerinnen und Schüler mit "ok" bestätigen sollten:

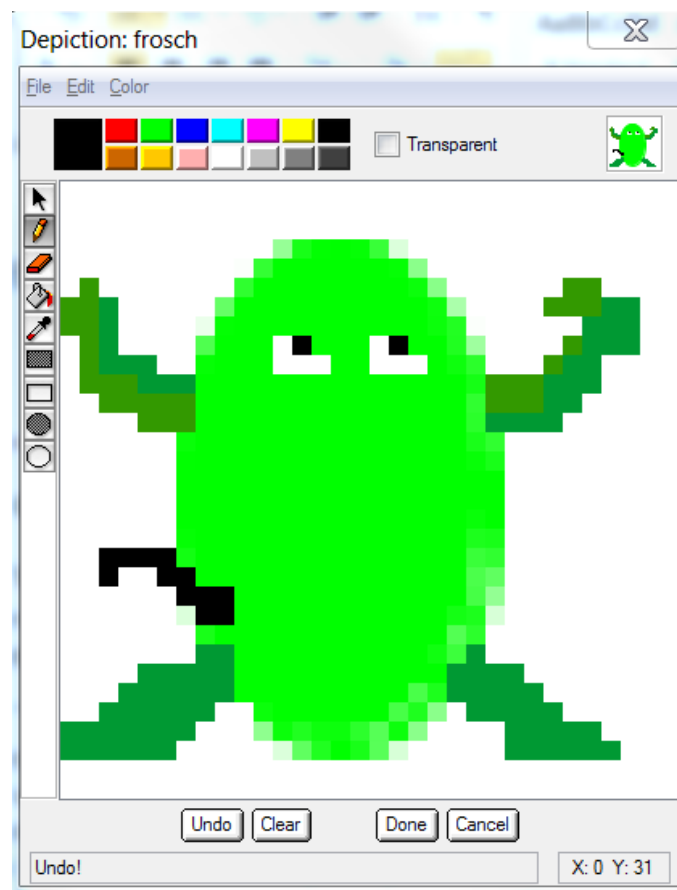


2. Design der Objekte

Das Design der "Spielfiguren" kann bei den einzelnen Schülerinnen und Schülern sehr unterschiedlich lange dauern. Generell sind die meisten Kinder beim 3D Design animierter als beim 2D Design. Es gibt verschiedene Möglichkeiten, das Tempo beim Erstellen der Objekte (agents) anzupassen.

- Man gibt den Schülerinnen und Schülern nach einiger Zeit eine Vorlage für die Objekte an die Hand.
- Man lässt die Lernenden zu Hause oder in Sonderstunden die Objekte kreieren.
- Einzelne Objekte können als vorgefertigte Bilder importiert werden (FILE -> IMPORT).

Hinweis: Beim Zeichnen der Objekte (agents) gibt es nur eine einmalige "undo" Möglichkeit, d.h. es kann nur jeweils ein Zeichenfehlschritt rückgängig gemacht werden. Gespeichert wird über die Eingabe "Done". Ein ganzes Bild kann über "Clear" gelöscht werden.



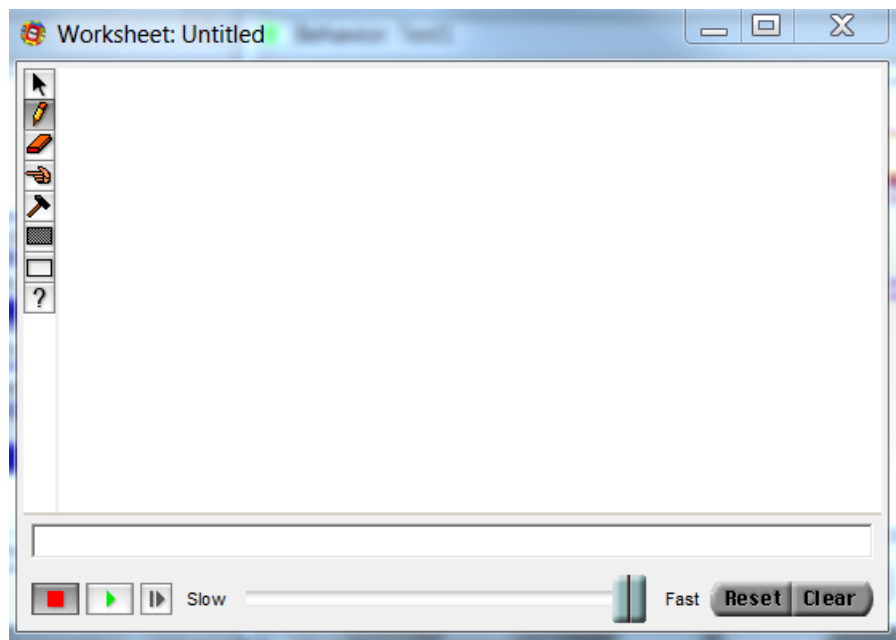
Sie entscheiden, wie viele Objekte die Klasse zu Beginn erstellt - es können nach und nach Objekte hinzukommen. Empfehlenswert ist es, mit dem Frosch, der Strasse und einem Fahrzeug zu beginnen.

3. Erstellen eines Spielfelds (worksheet) für das Spiel

Die Schülerinnen und Schüler platzieren erste Objekte auf ein Spielfeld (worksheet). Dazu muss im Programm folgende Navigation befolgt werden:

FILE -> NEW WORKSHEET

Nun erscheint ein neues "worksheet". Das Spielfeld kann vergrössert werden, aber es sollte nicht den ganzen Bildschirm einnehmen. Für die ersten Versuche und Tests ist es einfacher, wenn das Worksheet in seiner ursprünglichen Grösse bleibt. Oft passiert es nämlich am Anfang, dass die Dinge nicht gespeichert oder falsch gespeichert waren und dann muss alles neu auf dem Spielfeld platziert werden.



Es ist ratsam, ein mögliches Frogger-Spiel an die Tafel zu skizzieren, ohne den Lernenden ein Frogger-Spiel auf dem Computer zu zeigen. Durch die Skizze sehen sie, in welche Richtungen die Bewegungen verlaufen und können sich die Dimensionen besser vorstellen.

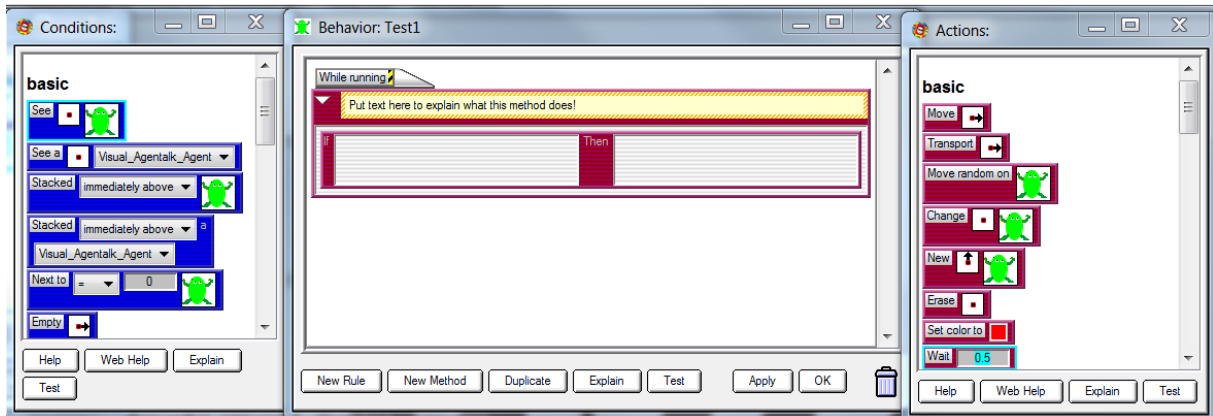
FEHLERQUELLE:

Eine häufige Fehlerquelle ist, dass mehrere Objekte übereinander gezeichnet werden. So kann es passieren, dass ein Frosch unter der Strasse sitzt und man sich später wundert, warum das Programm fehlerhaft funktioniert. Bei Zweifeln lieber Teile im Programm mit dem Radiergummi löschen.

4. Wir programmieren

Durch den Menü-Befehl EDIT BEHAVIOR in der Galerie, kann für das jeweilige Objekt die Programmierumgebung aufgerufen werden. Dabei öffnen sich 3 Dialogfenster:

1. Conditions (die Bedingungen)
2. Actions (die Aktionen)
3. Behavior (Eingabe der Programmierbefehle)

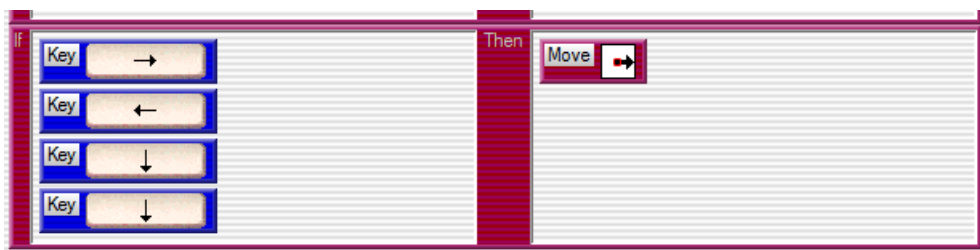


Über Drag&Drop werden die Befehle der Konditionen und Aktionen in die jeweiligen Fenster des Programmierdialogs gezogen. In den "If" Bereich gehören die blauen "conditions" und in den "Then" Bereich die roten "actions".

Die erste Überlegung ist an dieser Stelle, wie der Frosch seine benutzergesteuerte Bewegung bekommt. Dazu sollten sich Schülerinnen und Schüler die Möglichkeiten der beiden Befehlsboxen anschauen und eventuell kann man mittels "Vokabel-Cheat-sheet" eine deutsche Übersetzung der Begriffe besprechen.

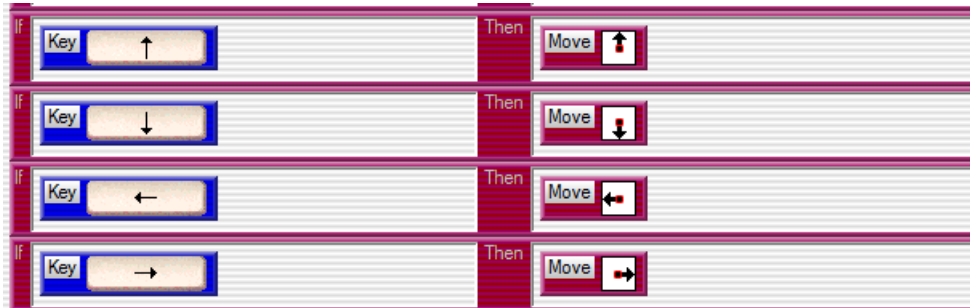
Als Ergebnis sollten Schülerinnen und Schüler die "key"-Bedingung (Tastatureingabe) in die *If*-Box einfügen und passend dazu den "move"- Befehl in die *Then*-Box. Es bleibt den Schülern überlassen, ob sie mit Pfeilen arbeiten möchten oder lieber die Kombination "W-A-S-D" als Richtungsangabe benutzen.

FEHLERQUELLE:



Im obigen Screenshot ist eine Programmierangabe gezeigt, die in sich widersprüchlich ist und nicht zum gewünschten Ergebnis führt. Dazu muss man wissen, dass Befehle innerhalb einer Regel (eines Kästchens) mit "und" verknüpft werden. So wäre dieses Programm wie folgt zu lesen:

Wenn die Pfeiltaste "->" und die Pfeiltaste "<-" und die Pfeiltaste "oben" und die Pfeiltaste "unten" alle gleichzeitig gedrückt sind, dann soll sich der Frosch nach rechts bewegen. Das entspricht nicht unserem Spielvorgang. Richtig müsste es wie im folgenden Screenshot dargestellt aussehen.



In diesem Programm sind die Regeln einzeln aufgeführt und durch ein "oder" verbunden. Hier würde das Programm wie folgt gelesen: Wenn die Pfeiltaste "oben" selektiert wird, dann bewegt sich der Frosch nach oben. Oder wenn die Pfeiltaste "unten" selektiert wird, dann bewegt sich der Frosch nach unten. Etc.