

Die Schildkröte lebt

Verschiedene Hochschulen bieten Unterrichtsprojekte zum Programmieren an. Von einem solchen Angebot profitieren zurzeit auch die Kinder zweier Klassen der Brunnmattschule in Basel. Sie lernen programmieren mit LOGO. **Von Werner Jundt.**

Vor dreissig Jahren war ich begeisterter Logo-Fan. Ich hatte «das» Buch gelesen: «Mindstorms - Kinder, Computer und neues Lernen» von Seymour Papert; es hatte mich gepackt. Die damals für mich neue konstruktivistische Sicht auf das Lernen einerseits, das mitgelieferte Instrument zur Umsetzung andererseits: Logo - eine Programmiersprache für Kinder. Also wurde ich wieder Kind - was habe ich nicht alles programmiert! - damals...

Solches geht mir durch den Kopf, als ich, vom Bahnhof kommend, im frühmorgendlichen Gundeldingerquartier unterwegs bin zum Schulhaus, in dem ich meine alte Bekannte, die Schildkröte, treffen werde.

Die sechste Klasse wird gegenwärtig in einer Doppelstunde pro Woche von zwei Studierenden der Uni Basel in Logo unterrichtet. Das Projekt ist auf zehn Wochen ausgelegt, finanziert wird es von der Hasler Stiftung. Zurzeit (Feb. 2017) nehmen im Raum Nordwestschweiz 18 Klassen an diesem Programm teil. Die Klassenlehrerin hat mit anderen Kolleginnen der Schule einen Einführungskurs besucht, sie hält sich aber im Hintergrund. Unterrichtsleitende sind die beiden Leute von der Uni.

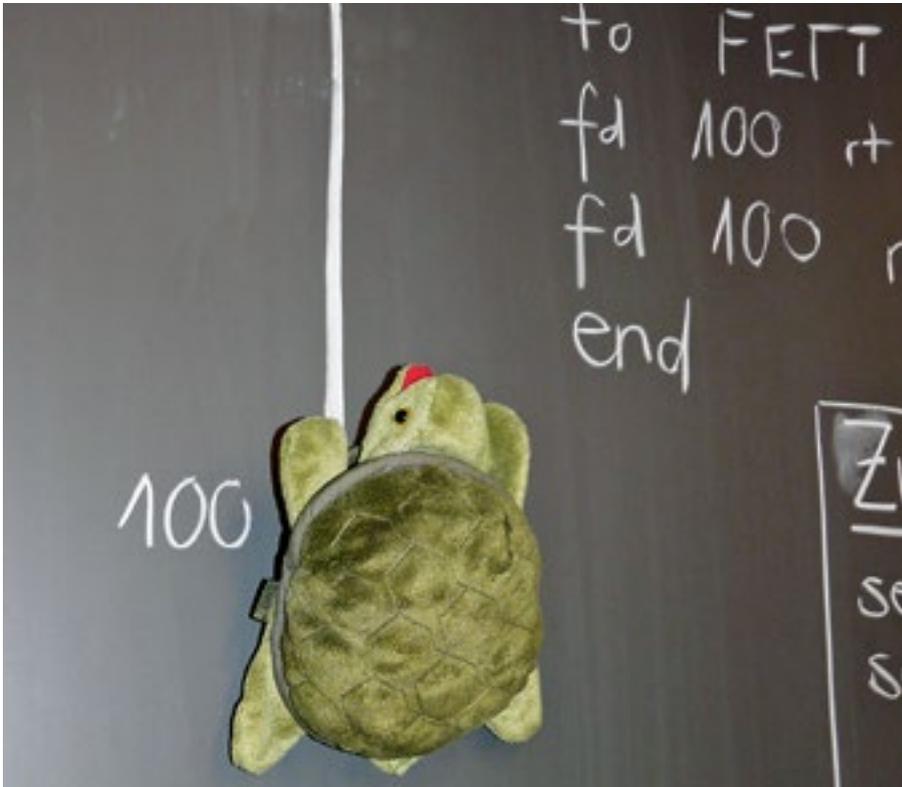
Die Schülerinnen und Schüler holen die Laptops aus der Rollkiste und starten sie auf. Sie sitzen zu viert oder sechst an Tischgruppen. Die Unterrichtsleiterin erfragt kurz den Stand

der Arbeiten und führt dann in eine nächste Gruppe von Aufgaben ein: «Es geht um den Befehl **FETT**.»

Die Instruktion erfolgt mithilfe einer Schildkröte. Die grosse Schwester der kleinen Schildkröte, welche die Kinder als Cursor auf ihrem Bildschirm steuern. Sie wissen: Mit dem

Befehl **fd 100** bringt man die Schildkröte dazu, sich 100 Pixel vorwärts zu bewegen und dabei einen 1-Pixel breiten Strich zu zeichnen. Sie verstehen auch gut, was mit **fd 100 rt 90 fd 1 rt 90 fd 100** erreicht wird: Ein Strich, dann rechtsum, 1 Pixel weiter, wieder rechtsum und ein Strich zurück. Das ergibt insgesamt einen





fetteren Strich. Und nicht vergessen, die Schildkröte wieder nach oben auszurichten: **rt 180**. Das kurze Programm erhält den Namen **FETT**.

Ich frage die Lehrerin, ob die englischen Abkürzungen keine Schwierigkeiten machen. «Nein. Das sind einfach Kürzel - und die Kinder lernen ja sowieso grad Englisch.» - «Geht das auch mit repeat?», fragt Timo. «Klar», sagt die Unterrichtsleiterin, «das geht immer, du wirst es gleich brauchen.» Timo hat wohl schon auf die nächsten Aufgaben geschaut und gesehen, dass er jetzt dann ganze Flächen einfärben soll. Und da hat er sich gedacht: Ein dicker Strich und noch ein dicker Strich und noch einer...

repeat ist ein mächtiger Befehl - und typisch für die Kraft, die man beim Programmieren hat. Einmal sagen, wie - und der Computer macht es, sooft man will. **repeat 100 (FETT)** und schon entsteht aus 100 Strichen ein ausgefülltes Quadrat. Und so, wie man sich aus dem **FETT** als Unterprogramm ein Programm **QUADRAT** konstruieren kann, gewinnt man aus diesem ein Programm **SCHACHBRETT** und daraus... Mit einem ganz kleinen Wortschatz lässt sich eine reiche Formenwelt darstellen. Und später muss es nicht mehr nur um Figuren gehen. Aber natürlich passieren auch Fehler, wenn die Sechstklässlerinnen und

Sechstklässler aus Quadraten weitere Figuren komponieren sollen. Viele Fehler. Programmieren ist eine optimale Schule für den Umgang mit Fehlern. Nirgendwo sonst erhalten Lernende in so kurzer Zeit so viele Rückmeldungen auf ihre Vorstellungen. Fehler machen wird selbstverständlich, und sie ausbügeln unverzichtbar. Und wenn's einfach nicht gehen will? Aline ist am Verzweifeln. Gottlob ist da noch der Unterrichtsleiter. «Steh mal neben das Pult», sagt er. «Du bist die Schildkröte. Ich lese, was du geschrieben hast, und du bewegst dich danach.»



fd 100 lt 90: 100 cm vor, 90° links

Aline führt ihre eigenen Befehle aus und merkt bald, wo's klemmt. Sie schreibt weiter. Die Schildkröte zeichnet. Mir fällt Jerome Bruner ein: Handlung - Bild - Sprache; und deren zentrale Bedeutung beim Aufbau abstrakter Konzepte.

Nach der Pause erlebe ich die gleiche Unterrichtseinheit in einer 5. Klasse. Ich suche mehr Gelegenheiten, mit einzelnen Kindern zu sprechen. Die Einstiegsfrage ist eine rhetorische: «Macht's Spass?» - «Klar», sagt Carla. - «Was macht Spass?» - «Computer, halt.» - «Ja klar, aber du machst ja vieles am Computer. Warum macht Logo Spass?» - «So Figuren zeichnen.» - «Das machst du ja auch ohne Computer, und erst noch viel schönere Figuren.» - Carla ist irritiert ob meiner Hartnäckigkeit. Sie schweigt. Ich warte. Sie überlegt, dann sagt sie mit einem genüsslichen Unterton: «Hier kann ich der Schildkröte etwas befehlen, und sie gehorcht mir.»

Ich denke, der grösste Gewinn, den Kinder beim Programmierenlernen davontragen, ist die Selbstwirksamkeitserfahrung. Die aber nur zum Preis konsequenter Überlegungen und präziser Formulierungen zu haben ist. Tricksen geht nicht: Das Gegenüber ist keine

”

«Die Schülerinnen und Schüler können einfache Problemstellungen analysieren, mögliche Lösungsverfahren beschreiben und in Programmen umsetzen», steht im Lehrplan 21.

Person, die sich erweichen lässt - aber auch keine Person, die irgendwelche Ideen unpassend findet. Heino möchte den ganzen Bildschirm schwarz einfärben. Mit **FETT** müsste das gehen, nur wie? Der Computer sagt dir nie, wie - aber er lässt dich alles ausprobieren, was dir in den Sinn kommt.

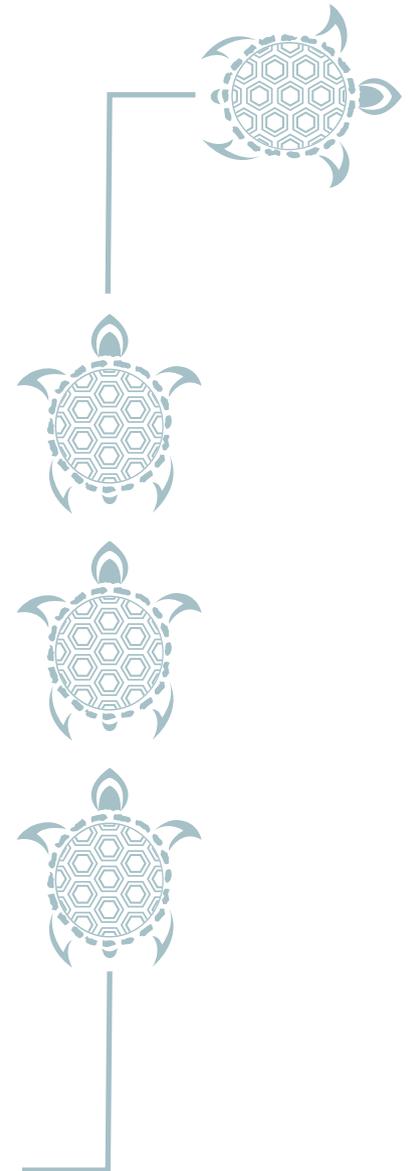
Leila ist an der Aufgabe, drei Quadrate ineinanderzuschachteln, unten links bündig. Sie schafft die Figur, ist aber unzufrieden. Den Unterrichtsleiter fragt sie, ob man denn wirklich nacheinander drei verschiedene grosse Quadrate zeichnen müsse. «Was denn sonst?», fragt er. Leila tut sich schwer damit, ihre Idee zu erklären: «Halt irgendwie schreiben, dass die Schildkröte versteht, dass sie Quadrate machen muss, aber immer grössere ... - und dann könnte man auch mehr als drei machen.» Ich tausche einen Blick mit dem Unterrichtsleiter. Wir schmunzeln. Da hat ein elfjähriges Mädchen grad die Variablen erfunden. «Doch das geht», sagt der Leiter. In drei Wochen kommen wir dazu.»

Es juckt mich, mit Leila weiterzuarbeiten. Aber so viel Differenzierung hat jetzt nicht Platz; und das liegt nicht an Logo. Im Gegenteil: Ich beobachte viele individuelle Tätigkeiten. Schnellere Schülerinnen und Schüler, die zwischen den Aufgaben im Heft noch rasch eine andere Idee ausprobieren. Ein Mädchen, das aus eigenem Antrieb eine schon gelöste Aufgabe noch eleganter lösen will, «mit einem Unterprogramm», sagt es mir.

Ich frage die Unterrichtsleiterin, ob Kompaktheit der Programme ein Thema sei.

«Kommt drauf an. Erstes Ziel ist, dass die Figuren so herauskommen, wie sie sollen. Von den Stärkeren darf man aber schon elegante Lösungen erwarten.» Alle Schülerinnen und Schüler folgen den Aufgaben im Heft. Aber wenn ich lese, was auf den Schirmen steht, sieht es überall ein wenig anders aus. Vordergründig lösen die Kinder Aufgaben, im Kern aber lernen sie Programmieren.

Auf dem Rückweg zum Bahnhof fällt mir ein, dass letztes Jahr kurz nacheinander Jerome Bruner und Seymour Papert gestorben sind. Nach dem eben Erlebten beschleicht mich leise Trauer. Nur kurz. Mein Haupteindruck von diesem Morgen: Die Schildkröte lebt! ■



”

... können Schleifen, bedingte Anweisungen und Parameter in selbst geschriebenen Computerprogrammen anwenden.

(LP21, Grundanspruch 2. Zyklus)