

Rekursion – Iteration Scratch

| | |
|---|---|
| Zielgruppe: | Schüler*innen am Ende der Sekundarstufe 1 oder in der Sekundarstufe 2 |
| Zeitrahmen: | 1-2 Unterrichtseinheiten |
| Fach: | Informatik, Mathematik |
| Lehrplanbezug: | <p>Digitale Grundbildung -Computational Thinking: <i>Mit Algorithmen arbeiten</i></p> <p>Mathematik – (Sekundarstufe 2 - 6. Klasse) <i>Potenzen</i></p> <ul style="list-style-type: none"> • <i>Potenzen (mit natürlichen, ganzen, rationalen bzw. reellen Exponenten) [...] definieren können; entsprechende Rechenregeln kennen und anwenden können,</i> <p><i>Folgen</i></p> <ul style="list-style-type: none"> • <i>Zahlenfolgen als auf \mathbb{N} bzw. \mathbb{N}^* definierte reelle Funktionen kennen (insbesondere arithmetische Folgen als lineare Funktionen und geometrische Folgen als Exponentialfunktionen); sie durch explizite und rekursive Bildungsgesetze darstellen und in außermathematischen Bereichen anwenden können,</i> • <i>Eigenschaften von Folgen kennen und untersuchen können (Monotonie, Beschränktheit, Grenzwert).</i> |
| Informatikkonzepte | Rekursion, Algorithmen |
| Typ/Art des Unterrichtsmaterials | Arbeitsaufträge mit Informationsmaterial |
| Benötigte Dateien: | <p>Allgemeine Informationen: RE_I_Rekursion-Iteration MO_I_Aktivitätsdiagramme</p> <p>Fibonacci: RE_I_Fibonacci RE_AA_Fibonacci_Scratch RE_LO_Fibonacci_Scratch_Iteration //Lösungen RE_LO_Fibonacci_Scratch_Rekursion</p> <p>Potenzen: RE_AA_Potenzen_Scratch RE_LO_Potenzen_Scratch_Iteration //Lösungen RE_LO_Potenzen_Scratch_Rekursion</p> |
| Utensilien: | Stift und Papier |
| Sozialform: | Einzel-, Partner- oder Gruppenarbeit |
| Lehrziele: | <p><i>Die Schüler*innen ...</i></p> <ul style="list-style-type: none"> • wissen, wie man Algorithmen ausführen kann, • können rekursive Abläufe nachvollziehen, • sind in der Lage, mit Python iterative und rekursive Abläufe darzustellen. |



| | |
|---------------------|--|
| Quellen: | Digitale Grundbildung BGBl. II Nr. 71/2018: https://www.ris.bka.gv.at/eli/bgbl/II/2018/71/20180419 (19.4.2018) Lehrplan AHS-Unterstufe https://bildung.bmbwf.gv.at/schulen/unterricht/lp/lp_ahs_unterstufe.html Alle Grafiken CC-BY-NC-SA Informatik-Werkstatt 2020 Bilder von Pixabay https://pixabay.com/de/ |
| Autor/innen: | Katharina Brugger |
| Lizenz: | CC-BY-NC-SA Informatik-Werkstatt AAU 2020 |

Hinweis:

Dieses Arbeitspaket ist mit anderen Paketen erweiter- und ergänzbar. Es gibt zu verschiedenen Programmiersprachen weitere Materialien, die man im Unterricht anschließend oder verschränkt behandeln kann. Es wird empfohlen, zuerst mit dem Arbeitspaket Rekursion-Iteration Pseudocode zu beginnen, welches unplugged die Algorithmen zu Fibonacci und Potenzen iterativ und rekursiv thematisiert. Danach kann man mit der Programmiersprache seiner Wahl weitermachen. In der Materialbörse kann man diese weiterführenden Arbeitspakete herunterladen (Namenskonvention: **Iteration-Rekursion *Programmiersprache***). Aufgrund dieser flexiblen Einteilung gibt es bei den Inhalten leichte Überschneidungen in den Dateien. Weiters ähneln sich die Strukturen der einzelnen Arbeitspakete, was zur leichteren Umsetzung der Inhalte im Unterricht verhelfen soll. Zu jeder Programmiersprache gibt es für jeden Algorithmus ein Aufgabenblatt mit bereitgestellten Lösungen. Weiters werden Informationsdateien zu den Inhalten bereitgestellt, welche die selbstständige Erarbeitung der Aufgabenblätter ermöglichen soll.

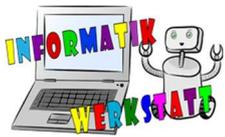
Vorbereitung:

Je nachdem, ob die Schüler*innen die Aufgaben einzeln, im Team oder in Kleingruppen erarbeiten sollen, werden die Aufgabenblätter und Informationsdateien (**RE_I_XxX** und **RE_AA_XxX**) in passender Anzahl ausgedruckt. Die Informationsdateien kann man bei Gruppenarbeiten auch laminieren, um sie wiederverwenden zu können. Die Aufgabenblätter werden im Allgemeinen von den Schüler*innen beschrieben, so soll jedes Kind ein Exemplar aller Aufgabenblätter erhalten. Die Lösungen (**RE_LO_XxX**) können zur selbstständigen Kontrolle bereitgestellt werden oder auch als Orientierung für die Lehrperson bei der Korrektur und Bewertung der Abgaben herangezogen werden.

Einsatz/Handhabung:

Die Schüler*innen sollen sich selbstständig in das Thema einarbeiten können. Sie erhalten alle benötigten Informationen von den Informationsblättern. Man kann als Lehrperson auch eine Einführungsphase einplanen, wo gemeinsam über das Thema gesprochen wird. Wichtig ist, dass die Kinder den Begriff Algorithmus bereits kennen und in der thematisierten Programmiersprache gearbeitet haben. Die Arbeitsaufträge gehen nicht auf alle Grundlagen ein, da dies bereits als Vorwissen vorausgesetzt wird.

Als Lehrperson ist man in der Unterrichtseinheit für die Klärung von Fragen und Problemen zuständig. Weiters können besonders schnelle Schüler*innen oder Gruppen, dessen Ergebnisse sauber ausgearbeitet sind, als Tutor*innen für die restliche Klasse fungieren.



Tipps und Veränderungen

Bei manchen Aufgaben werden Aktivitätsdiagramme verwendet. Falls die Kinder diese Diagramme nicht kennen, haben sie ein Informationsblatt zur Verfügung. Man kann aber auch davor in einer oder mehreren Unterrichtseinheiten das Informatikkonzept der Modellierung und in weiterer Folge auch Aktivitätsdiagramme behandeln. Hier gibt es in der Materialbörse unter <https://www.rfdz-informatik.at/materialboerse/> ebenfalls Materialien, die dafür herangezogen werden können.