

Computing und Computational Thinking

Unter **Computing** versteht man die Verwendung eines Computersystems zur Durchführung von Berechnungen oder der Verarbeitung von Daten. Die Erstellung von Programmen für den Computer fällt auch unter dem Begriff Computing.

Computational Thinking (Informatisches Denken) hilft komplexe Probleme zu verstehen und zu lösen. Auch sollen Probleme und dessen Lösung von Computern verstanden werden, bzw. müssen sie in einer Computersprache formulierbar sein. Dazu ist Computational Thinking notwendig!

Computational Thinking setzt sich aus mehreren Teilbereichen zusammen:

- Komplexe (vielfältige) Sachverhalte in verstehbare Teilprobleme zu zerlegen, dass diese einfacher lösbar sind.
- Ähnliche Muster zu erkennen, Muster können schon in anderen Problemlösungen bekannt sein.
- Abstraktion: Wesentliches von Unwesentlichem zu unterscheiden. Was ist notwendig? Dies vereinfacht das Problem.
- Algorithmen (Folge von Schritten): Das Problem wird verstanden. Jetzt können Algorithmen entwickelt werden, die zur Problemlösung führen.
- Evaluation (Überprüfung) des Algorithmus

Computational Thinking ist nicht nur in Zusammenhang mit Computern wichtig, sondern spielt auch im Alltag eine große Rolle:

- Wir müssen Arbeiten erledigen, die in einem bestimmten Ablauf erfolgen.
- Diese Arbeiten sollen möglichst schnell, effizient und auch automatisiert ablaufen.

Beispiele:

- Kochen: Arbeitsschritte folgen aufeinander, aber es finden auch welche parallel statt – während das Gemüse gegart wird, wird der Salat angerichtet,
- ein Gegenstand wird repariert,
- ein Möbelstück zusammengebaut,
- ein Produkt entwickelt usw.

Computational Thinking ist somit eine Fähigkeit, die jeder benötigt!

Computational Thinking verwendet vier grundlegende Problemlösungstechniken.

1. Mustererkennung (Pattern Recognition) und Generalisierung (Verallgemeinerung)

Ein Muster ist eine **wiedererkennbare Anordnung**. Beispiele sind ein Fliesenmuster, ein Teppich mit sich immer wiederholenden Motiven oder Musikstücke.

Auch in der Mathematik können Muster entdeckt werden:

Wie heißt die nächste Zahl: 1,2,4,8,16,.....

Begründe dies:

Ein Beispiel könnte ein Computerprogramm für Scratch sein: Gehe 100, drehe dich um 90°, gehe 100, drehe dich um 90°, usw., das viermal: Ein Rechteck entsteht.

Einfacher - führe 4-mal aus: gehe 100 und drehe dich um 90°

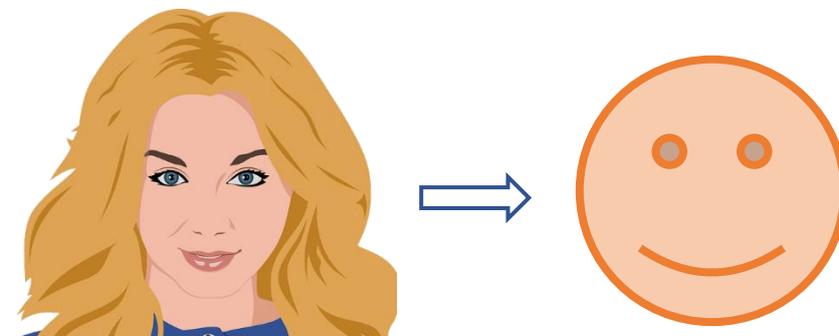
Generalisierung: Werden Ähnlichkeiten bzw. Muster in Problemen erkannt, können **Lösungswege übernommen** werden.

2. Abstraktion

Bei der Abstraktion **filtert man unnötige Details heraus** und konzentriert sich auf die Informationen, die für die Lösung des Problems **wichtig** sind.

Bei der Aufgabe den richtigen Weg in die Schule zu gehen, ist es nicht wichtig, ob man dabei Musik hört. Es ist aber wichtig, immer die richtige Straße einzubiegen, damit man an das Ziel kommt, bzw. keine Umwege macht.

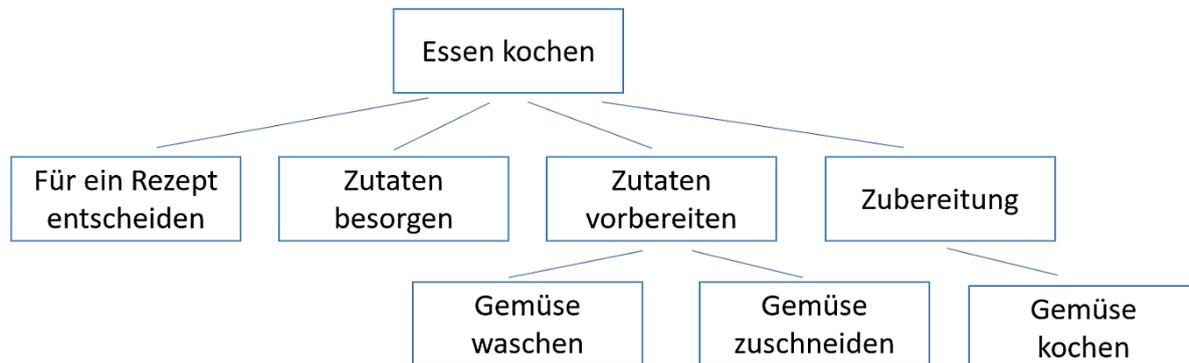
Im folgenden Beispiel wurden alle unnötigen Details des lachenden Gesichts entfernt, so dass nur die Informationen übrigbleiben, die für die Lösung des Problems wichtig sind – hier ist es ein lachendes Gesicht.



3. Problemzerlegung

Problemzerlegung bedeutet die **Zerlegung eines komplexen Problems in kleinere Probleme**, die einfacher und leichter zu verstehen sind.

Auch die kleineren Probleme können immer weiter in noch kleinere Probleme zerlegt werden, bis sie leicht verständlich und lösbar sind. Das Beispiel *Kochen von Essen* zeigt, wie ein komplexes Problem in immer kleinere Teile und Probleme zerlegt werden kann.



4. Algorithmus

Ein Algorithmus besteht aus einer Folge von Anweisungen zur Lösung einer Aufgabe.

Ein Algorithmus könnten Schritte von Berechnungen, die ein Computer durchführen soll, oder hier, aus dem Alltag gegriffen, die Anleitung eines Kochrezeptes sein.

Ein möglicher Algorithmus zur Zubereitung von Salzburger Nockerl enthält folgende Schritte:

1. Zutaten zusammenstellen: 5 Eier, 3 EL Zucker, 1 EL Vanillezucker, 1 EL Mehl
2. Eiklar schaumig rühren, Zucker dazugeben und rühren
3. 3 Eidotter und Mehl darunterheben
4. 20 Minuten bei 80° im Backrohr

Zwei zusätzliche Ansätze werden oft zu Computational Thinking hinzugefügt: Evaluierung und Verallgemeinerung:

5. Evaluierung (Evaluation)

Die Evaluierung beinhaltet die Überprüfung des Entwurfs eines Produktes oder eines Algorithmus. Dabei wird überprüft, ob der gewünschte Zweck damit erfüllt, beziehungsweise ob das Problem damit gelöst wird.

Begriffe kurz erklärt

Computing	Durchführung von Berechnungen, Verarbeitung von Daten
Computational Thinking	Probleme werden analysiert, Lösungen und Lösungswege gefunden
Mustererkennung	Wiederkehrende Anordnungen werden erkannt.
Problemzerlegung	Zerlegung von komplexen (vielfältigen) Sachverhalten in kleinere Teilprobleme. Damit wird das Problem verständlicher und leichter lösbar.
Komplexes Problem	Bedeutet vielfältig, vielschichtig, umfangreich...
Algorithmen	Folge von Schritten. Eindeutige Handlungsvorschrift oder Anweisung. Nachdem das Problem verstanden worden ist, werden Algorithmen entwickelt, die zur Problemlösung führen.
Evaluation, Validierung	Der Algorithmus wird überprüft. Auch wird überprüft, ob der gewünschte Zweck erreicht worden ist.
Eindeutige Handlungsvorschrift	Die Anleitung ist klar und nicht mehrdeutig: Z. B. wird im Kochrezept 300 Gramm Mehl und nicht eine Menge nach Belieben als Zutat angegeben.

Beantworte folgende Fragen, bzw. ergänze:

1. Bei der Abstraktion werden unnötige herausgefiltert. Man konzentriert sich nur auf die Informationen, die für die Lösung des Problems wichtig sind.

2. Ein Algorithmus könnte zum Beispiel einrezept sein.

4. Unter Problemzerlegung versteht man die Zerlegung eines Problems in Probleme, die einfacher und leichter zu verstehen sind.

4. Unter versteht man die Durchführung von Berechnungen oder die Verarbeitung von Daten, insbesondere unter Verwendung eines Computersystems.



5. Wie nennt man die Überprüfung, ob der gewünschte Zweck erreicht worden ist? A:

Fotografiere den QR-Code oder aktiviere den Link. Löse das Quiz: