

Problemlösung

Wie lösen wir Probleme systematisch?

Pólya und Conway (2004) beschreiben in ihrem Aufsatz *How to Solve It* einen weit verbreiteten Ansatz zur Problemlösung. Dieser basiert auf vier Schritten, die als wiederkehrender Kreislauf angewendet werden können. Obwohl Pólya und Conway (2004) diesen Ansatz ursprünglich für mathematische Probleme entwickelten, lässt er sich auch auf andere Bereiche – insbesondere die Informatik – übertragen. Folgende Schritte beschreiben beide in ihrem Aufsatz:

1. Das Problem verstehen
2. Einen Plan zur Lösung erstellen
3. Den Plan umsetzen
4. Die Lösung reflektieren und verbessern

Während alle wichtig sind, konzentrieren wir uns auf den ersten Schritt: das Problem zu verstehen. Bei der Lösung eines Problems mithilfe eines Computers müssen wir diesen Schritt in zwei Stufen unterteilen. Die erste Stufe beinhaltet eine gründliche Auseinandersetzung mit dem Problem in der realen Welt, um dessen Kern zu erfassen. Diese Grundvoraussetzung wird oft vernachlässigt. Erst wenn ein klares Verständnis des Problems vorhanden ist, können wir zur zweiten Stufe übergehen: der Übersetzung des Problems in ein Modell, das die computerrelevanten Aspekte des Problems erfasst und beschreibt. Wir führen dafür unten das EVA-Modell ein.

Der zweite Schritt im Ansatz von Pólya und Conway (2004) ist die Erstellung eines Plans für die Lösung des Problems. In der Informatik besteht dieser Plan aus einer Reihe von Anweisungen, die in einer bestimmten Reihenfolge ausgeführt werden müssen. Dies lässt sich gut mit einer Wegbeschreibung vergleichen: “Fahre 500 Meter geradeaus, biege an der Kreuzung links ab, fahre bis zum zweiten Kreisverkehr und nimm dort die zweite Ausfahrt. Nach 200 Metern findest du dein Ziel auf der linken Seite.” Eine solche schrittweise Anleitung nennen wir in der Informatik Algorithmus.

Nachdem wir einen Plan zur Lösung des Problems erarbeitet und in geeigneter Form aufgeschrieben haben, müssen wir ihn im dritten Schritt umsetzen. In der Informatik bedeutet dies, den Lösungsweg einem Computer beizubringen. Dies erreichen wir durch das Schreiben eines Programms in einer Programmiersprache wie etwa Python. Das Programmieren zu erlernen ist eines unserer Ziele.

Es wäre überraschend, wenn wir an unserer Lösung keine Verbesserungsmöglichkeiten fänden. Dies gilt besonders für die Informatik und unsere entwickelten Programme. Häufig entdecken wir Fehler, die wir beheben müssen, oder stoßen auf Sonderfälle des Problems, die wir im ersten Entwurf nicht bedacht haben. Da Informatiker meist Programme für andere Menschen entwickeln, erhalten wir von den Nutzern wertvolles Feedback mit konkreten Verbesserungsvorschlägen. Die Reflexion und Verbesserung unserer Lösung ist der letzte Schritt im Ansatz von Pólya und Conway (2004). Wie die Pfeile in Abbildung ?? zeigen, führt dieser letzte Schritt uns oft zurück zum ersten Schritt. Dadurch gewinnen wir ein tieferes Verständnis des Problems, was wiederum Änderungen am Plan und der Umsetzung erfordert. Gerade in der Softwareentwicklung ist man eigentlich nie wirklich fertig.

Polya's approach to problem-solving

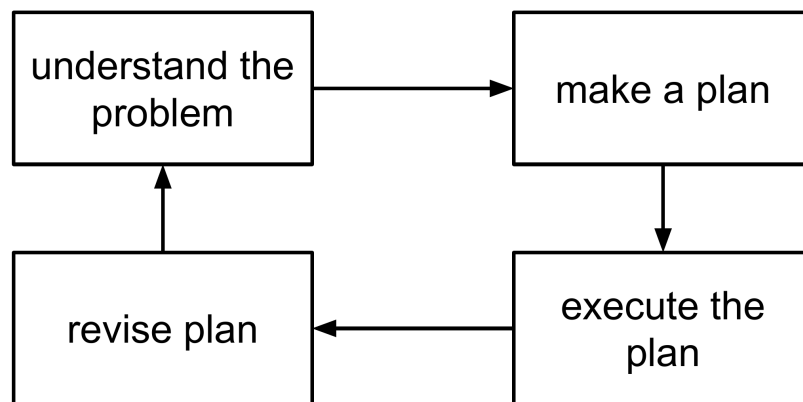


Abbildung 1: Der Ansatz von Polya zum Lösen von Problemen.

Bevor wir uns mit dem oben genannten EVA-Modell beschäftigen, das uns hilft, Probleme adäquat für Computer zu beschreiben, wollen wir einen kurzen Blick auf eine andere Frage werfen: Warum überhaupt Computer zur Problemlösung?

Warum sind Computer beim Lösen von Problemen nützlich?

Der wichtigste Grund für die Nutzung von Computern ist das Lösen von Problemen. Ob wir eine Route mit Google Maps planen, Online-Bestellungen bei DHL verfolgen oder eine KI wie ChatGPT um eine Empfehlung bitten – überall lösen Computer Probleme. Warum? Weil

Computer zwei Eigenschaften besitzen, die für viele Probleme und deren Lösung vorteilhaft sind:

1. Computer machen keine Fehler. Wenn wir einem Computer einen Lösungsweg beibringen, wendet er ihn fehlerfrei auf neue Probleme an.
2. Computer sind unglaublich schnell. Ob einfache Schritte, komplexe Berechnungen oder die Verarbeitung großer Datenmengen – Computer lösen Probleme in einem Bruchteil der Zeit, die wir Menschen benötigen würden.

Diese beiden Eigenschaften ermöglichen es uns, mit Computern besonders solche Probleme effizient zu lösen, die wiederkehrend und in großer Zahl auftreten. Wir sprechen dann von **Automatisierung**.

In diesem Abschnitt lernen wir, wie Computer Probleme strukturieren und lösen. Um den Begriff des Problems besser zu verstehen und seine Bedeutung im Kontext von Computern einzugrenzen, führen wir zunächst ein einfaches Modell ein.

Wie stellen wir Probleme für Computer dar?

Ein Programm zu schreiben stellt eine komplexe Herausforderung dar. Beim Lesen einer Aufgabenstellung, die man als Programm umsetzen soll, kann man sich schnell überfordert fühlen. Die zentrale Frage lautet: Wie nähern wir uns dieser Aufgabe systematisch an?

Ein bewährter Ansatz für komplexe Situationen ist die Vereinfachung. Auch wenn wir das Problem selbst nicht vereinfachen können, können wir es durch eine strukturierte Herangehensweise besser verstehen und handhabbarer machen. Die Verwendung von Modellen ist dafür ein geeigneter Weg.

Modelle zielen darauf ab, die wesentlichen Aspekte der realen Welt hervorzuheben und unwichtige Details auszublenden. Da dies zunächst abstrakt klingen mag, werden wir es anhand eines Modells veranschaulichen, das uns durch das gesamte Buch begleiten wird: das Eingabe-Verarbeitung-Ausgabe-Modell, kurz EVA-Modell.

Das Eingabe-Verarbeitung-Ausgabe-Modell

Das Eingabe-Verarbeitung-Ausgabe-Modell (EVA-Modell, s. Abbildung ??) ist ein wichtiges Modell in der Informatik. Es erklärt die Arbeitsweise von Computern auf vereinfachte Weise und beinhaltet nur die nötigsten Elemente. Konkret zeigt das Modell, wie Computer Probleme lösen und welche drei Elemente wir dabei betrachten müssen: Computer benötigen **(1) Eingabedaten**, die sie durch einen definierten **(2) Verarbeitungsprozess** in gewünschte **(3) Ausgabedaten** umwandeln.