

Lernziele

Der Unterricht, dem Ihre Schülerinnen und Schüler folgen, ist mit einer Reihe von Lernzielen verknüpft. Kompetenzen des 21. Jahrhunderts und Medienkompetenz spielen dabei eine immer größere Rolle. In der Abb. sehen Sie, in welche Bereiche sich die "Kompetenzen des 21. Jahrhunderts" unterteilen lassen. Besonders hervorzuheben sind hierbei die vier Schlüsselqualifikationen Kollaboration, Kommunikation, Kritisches Denken und Kreativität, die so genannten 4K-Kompetenzen.

Robotik und Programmierung, wie sie mit Ozobot durchgeführt werden, passen perfekt zum „Computerbasierten Denken“. Die Lernziele für diese Kategorie sind nachfolgend ausführlich beschrieben und werden mit den Lektionen von Ozobot perfekt erreicht.

Wenn Sie sich für jede Unterrichtsstunde die entsprechende Karte für Lehrkräfte ansehen, finden Sie dort eine Reihe von Lernzielen. Dies sind die Lernziele, die mit der jeweiligen Lektion erreicht werden sollen.

1. Probleme (um)formulieren

Ein Problem so (um)formulieren, dass ein Computer es lösen kann.

- 1.1 Nach Möglichkeiten suchen, um Probleme mit einem Computer zu lösen.
- 1.2 Rückblick auf die getroffenen Entscheidungen und gefundenen Lösungen.
- 1.3 Eine Problemstellung so (um)formulieren, dass ein Computer sie umsetzen kann.

2. Daten sammeln

Sammeln von Daten zu Codeerstellung.

- 2.1 Verwertbare Daten aus allen vorliegenden Daten sammeln.
- 2.2 Aus dem Ergebnis eine Datensammlung generieren.

3. Daten analysieren

Daten logisch organisieren und verstehen, Muster erkennen und Schlussfolgerungen ziehen.

- 3.1 Erkennen, was ein einfaches Muster ist.
- 3.2 Weiterführen und Erstellen von Mustern in konkreten Situationen.
- 3.3 Muster in formalen abstrakten Darstellungen der Realität erkennen.
- 3.4 Die Anwesenheit von Mustern in konkret-formalen und formal-abstrakten Situationen prüfen.
- 3.5 Muster finden, die bestimmte Abläufe vereinfachen.

4. Daten visualisieren

Daten in geeigneter Form darstellen.

- 4.1 Verschiedene Arten von Daten in eine für den Computer geeignete Form bringen (z.B. Binärcodes).

5. Problemzerlegung

Ein Problem oder eine Aufgabe in kleinere Teilprobleme oder Unteraufgaben zerlegen, um die Komplexität zu reduzieren.

- 5.1 Eine leichte Aufgabe in Teilaufgaben unterteilen.
- 5.2 (Teil)-Aufgaben in eine logische Reihenfolge bringen.
- 5.3 Teile eines großen Objekts als Teile eines größeren Ganzen einordnen.
- 5.4 Eine Aufgabe in mehrere Teilaufgaben zerlegen.
- 5.5 Zerlegen einer größeren und komplexeren Aufgabe in mehrere Teilaufgaben.
- 5.6 Überprüfen, ob bei der Durchführung von Teilaufgaben ein wichtiger Teil übersehen oder vergessen wurde.

6. Automatisierung

Verständnis für die Möglichkeiten des Computers zur Ausführung sich wiederholender oder monotoner Aufgaben. Verwendung des Computers zu Verarbeitung/Bearbeitung von Daten.

- 6.1 Wiederholung von Aufgaben in verschiedenen Situationen erkennen.
- 6.2 Bewusstsein für die Fähigkeit des Computers, Aufgaben endlos zu wiederholen.
- 6.3 Beispiele für sich wiederholende Aufgaben kennen, für die ein Computer verwendet wird.
- 6.4 Benennung von Beispielen für automatisierte Systeme im Alltag.
- 6.5 Benennung von Vor- und Nachteilen der Automatisierung von Aufgaben.

7. Algorithmen und Abläufe

Erklären, wie Algorithmen als eine Reihe geordneter Anweisungen oder Regeln funktionieren, die Schritt für Schritt ausgeführt werden, um ein Problem zu lösen oder ein Ziel zu erreichen. Entwerfen einfacher Algorithmen und Korrektur der enthaltenen Fehler.

- 7.1 Eine Aufgabe ausführen, indem eine Reihe von Aktionen Schritt für Schritt ausgeführt wird.
- 7.2 Befolgen logischer Abfolgen von Anweisungen.
- 7.3 Erteilen einer Reihe von Anweisungen anderen gegenüber (verbal oder durch Symbole), um eine bestimmte Aufgabe auszuführen.
- 7.4 (Teil-)Aufgaben in eine logische Reihenfolge bringen.
- 7.5 Die Bedeutung des Begriffs „Algorithmus“ kennenlernen.
- 7.6 Verständnis dafür, dass ein Algorithmus eine Abfolge von Anweisungen ist, die zu einem bestimmten Ergebnis führt.
- 7.7 Verständnis dafür, dass Computerprogramme etwas tun, indem sie präzise und eindeutige Anweisungen befolgen.
- 7.8 Erstellen eines einfachen Algorithmus in einer konkreten Situation mit einem fest vorgegebenen Satz von Anweisungen.
- 7.9 Darstellung und Kommunikation eines Algorithmus durch Codes und Symbole.
- 7.10 Erstellen einfacher Abfolgen von Anweisungen, die zu einem bestimmten Ziel führen.
- 7.11 Fehler in Algorithmen durch logisches Denken finden und beheben (Debugging).
- 7.12 Sammeln praktischer Erfahrungen in einer Programmierumgebung (Software).
- 7.13 Beschreiben einer Wiederholungsschleife mit einer festen Anzahl von Wiederholungen.
- 7.14 Das Konzept des Algorithmus mit Alltagssituationen verknüpfen.
- 7.15 Entwickeln und Schreiben eines einfachen Computercodes.
- 7.16 Verwendung einer Wenn-dann-Konstruktion bei der Beschreibung von Schritten in einem Prozess.
- 7.17 Vorhersage des Verhaltens bei der Nutzung einfacher (Computer-)Programme durch logisches Denken.