

Diagnostische Leitfragen und entwicklungsorientierte Fördermaßnahmen

Stand: 29.04.2019

Lernbereich 4.2: Programmieren – Abläufe und Algorithmen

| Diagnostische Leitfragen | Entwicklungsorientierte Fördermaßnahmen |
|---|---|
| Handlungsabläufe aus der Lebenswelt | |
| <p>Kann die Schülerin bzw. der Schüler</p> <ul style="list-style-type: none"> • sequenzielle Handlungsabläufe aus der Lebenswelt beobachten und beschreiben? • eine Handlung in einzelne und elementare Teilschritte untergliedern? • algorithmische Eigenschaften von Teilschritten untersuchen? • Anweisungen auf ihre Allgemeingültigkeit, Eindeutigkeit, Ausführbarkeit und Endlichkeit hin überprüfen? | <ul style="list-style-type: none"> • Unterstützung beim Finden programmierter Abläufe aus der unmittelbaren Lebenswelt geben • Gliederung von Handlungen in einzelne Teilschritte, wie bei einer Vorgangsbeschreibung • Handlungen praktisch ausführen lassen (z. B. Stift spitzen, Mäppchen öffnen, Arbeitsblätter abheften, ...) und diese in Teilschritten verbalisieren • Anhand einer Checkliste die Teilschritte eines Handlungsablaufs auf die Eigenschaften endlich, eindeutig, ausführbar und allgemeingültig überprüfen, z. B. „Gehe vorwärts“ ist nicht endlich im Vergleich zu „Gehe fünf Schritte vorwärts“) |
| Klare Handlungsanweisungen | |
| <p>Kann die Schülerin bzw. der Schüler</p> <ul style="list-style-type: none"> • klare Handlungsanweisungen formulieren? • im Rollenspiel nach klaren Anweisungen handeln? • Notwendigkeit klarer Handlungsanweisungen erfassen? • Handlungsanweisungen nach vorgegebenem Muster mit Programmiersprache verschriften? | <ul style="list-style-type: none"> • einen Menschroboter (Lehrer- oder Schülerroboter) im Rollenspiel mit und ohne Sprechen einen Parcours ablaufen lassen • Befehle schriftlich festhalten (z. B. Befehlsblöcke, Pseudocode, ...) • Einfache, schriftlich dokumentierte Befehle ausführen • Handlungsanweisungen unter Verwendung vorgefertigter Befehlsblöcke zusammensetzen |

| Objektorientierte Arbeitsweise | |
|--|--|
| <p>Kann die Schülerin bzw. der Schüler</p> <ul style="list-style-type: none"> Eigenschaften eines Objekts erkennen und benennen? Objekten bestimmte Eigenschaften zuweisen und ihnen Anweisungen geben? verschiedene Objekte miteinander interagieren lassen? | <ul style="list-style-type: none"> Unterschiedliche Eigenschaften von beliebigen Objekten aus der Lebenswelt zeigen, verbalisieren und gegenüberstellen Startpunkt im Rollenspiel festlegen, Blickrichtung benennen, Ausgangskonfiguration wiederherstellen Im Rollenspiel zwei Menschroboter interagieren lassen (z. B. sich grüßen) |
| Objekte erstellen und verändern | |
| <p>Kann die Schülerin bzw. der Schüler</p> <ul style="list-style-type: none"> in einer passenden Entwicklungsumgebung Objekte erstellen? | <ul style="list-style-type: none"> Unterstützungssysteme für Zugriff auf interne Objektbibliotheken oder externe Dateiquellen etablieren (z. B. Hilfskärtchen, Tutorials, ...) |
| <ul style="list-style-type: none"> Eigenschaften von Objekten (z. B. Raumlage, Geschwindigkeit, Farbe, Größe) in einer pädagogischen Entwicklungsumgebung verändern? | <ul style="list-style-type: none"> Unterstützungssysteme (z. B. Erklärvideo) beim Auffinden und Anpassen der jeweiligen Eigenschaftsdialoge im Programm (z. B. Kostüme, Drehtyp, ...) |
| Algorithmische Strukturen und Variablen verwenden | |
| <p>Kann die Schülerin bzw. der Schüler</p> <ul style="list-style-type: none"> zur Lösung einer gegebenen Problemstellung zielführende Abläufe modellieren? modellierte Abläufe in eine pädagogische Entwicklungsumgebung implementieren? algorithmische Strukturen und Variablen verwenden? | <ul style="list-style-type: none"> Verwendung von vorgefertigten Befehlskarten, Pseudocode, usw. Verwendung von vorstrukturierten Hilfsdateien zu entsprechenden Problemstellungen (z. B. Rücksetzen auf die Ausgangsposition). Verwendung von vorgefertigten Befehlsblöcken und Erstellung von Struktogrammen. |

Programme testen und optimieren

Kann die Schülerin bzw. der Schüler

- | | |
|---|--|
| <ul style="list-style-type: none"> • überprüfen, ob ein Programmablauf zum gewünschten Ziel führt? •einen Fehler im Programmablauf der entsprechenden Stelle im Programmiercode zuordnen? •ein Programm im Hinblick auf Übersichtlichkeit, Lesbarkeit, Nachvollziehbarkeit, Arbeitsökonomie, Komplexität, Ressourcenschonung usw. überprüfen und verbessern? | <ul style="list-style-type: none"> •Den gewünschten Ablauf vorab (evtl. In Teilschritten) verbalisieren, um ihn mit dem implementierten Programm zu vergleichen (z. B. mit Partner) •Einen Testdurchlauf von Einzelschritten oder Teilabschnitten mit dem gewünschten Ergebnis vergleichen. •Das Programm dem Partner oder der Gruppe präsentieren, alternative Lösungsvorschläge annehmen und Verbesserungen umsetzen. |
|---|--|