

HOPP ITs



## Lehrerhandreichung

Ein Spiel von Christopher Bednorz,  
Daniel Stoll und Jasmin Mangei

Auf <https://www.hopp-foundation.de/foerderangebot-schule/informatik/informatikspiele/code-mainia/> steht die Lehrerhandreichung zum Download bereit.



- › **Lernbereich:** Algorithmik und Daten
- › **Klassenstufe:** 8–13
- › **Thema:** Variablenbegriff und algorithmische Grundkonzepte wie Schleifen und Verzweigungen
- › **Spieler pro Spielset:** 3–5
- › Geeignet als Einstieg in die Programmierung oder zur vertiefenden Wiederholung
- › **Zeitaufwand:** 7–15 Minuten (davon 5 Minuten Einarbeitungszeit)

»Code Mainia« ist ein Kartenspiel für Schülerinnen und Schüler (SuS), bei dem es darum geht, einen Programmcode sequentiell zu legen und auszuwerten. Die SuS verfolgen das Ziel, den Code beim Ablegen der Karten möglichst schnell zu erfassen und im Kopf auszuführen. Der Code ist mit der Program-

miersprache Java formuliert und kompatibel mit C++. Das Spiel kann in zwei Varianten gespielt werden: In der Variante *Grundspiel* sollen die SuS die Anweisungen auf den Karten erklären und so eine Diskussion mit ihren Mitspielern beginnen, in der Variante *Dynamische Erweiterung* treten sie in einen Wettstreit miteinander ein.

## HINWEISE ZUR KOMPETENZVERMITTLUNG

### Fachbezogene Kompetenzen

- › Das Spiel »Code Mainia« lässt sich der Leitidee »Algorithmen und Daten« in den Bildungsstandards für Informatik für die Kursstufe an Gymnasien in Baden-Württemberg zuordnen.
- › Bei diesem Kartenspiel lernen die SuS das Konzept der Variablen und den Umgang damit kennen. Die unterschiedlichen Farben der Spielkarten signalisie-

ren die erforderliche fachliche Kompetenz. Somit ist es möglich, »Code Mainia« in unterschiedlichen Unterrichtskontexten passend zum aktuellen Lernstand einzusetzen.

- › Der Programmcode auf den Karten beinhaltet Zuweisungen, Vergleiche, einfache Operationen (Addition, Subtraktion, Multiplikation und Division) und je nach Variante auch den Modulo-Operator, Kontrollstrukturen und Schleifen. Neben dem algorithmischen Denken fördert das Spiel die schnelle Auffassung, Interpretation und Auswertung eines Programmcodes.
- › Die Spielanleitung des *Grundspiels* ist auch als Struktogramm verfügbar. Dadurch können die SuS eine weitere Herangehensweise an Problemformulierungen üben bzw. vertiefen.

### Prozessbezogene Kompetenzen

- › »Code Mainia« fördert die Sozial- und Selbstkompetenz der SuS. Die Spieler müssen zu jedem Zeitpunkt des Spiels gerade vollzogene Anweisungen verste-

hen und erklären können. Die SuS besprechen das generierte Programm miteinander und diskutieren den Ablauf in Einzelschritten.

- › Unterschiedliche Ergebnisse werden im Gruppengespräch geklärt. Beim *Grundspiel* spielt Zeit keine Rolle. So können die SuS die Einzelanweisungen im eigenen Lerntempo erklären oder nachvollziehen.
- › Im Spielverlauf wird der aktuelle Wert der Variablen mehrfach abgefragt; wird der richtige Wert genannt, fördert dies das Selbstvertrauen in die eigene Fähigkeit, einen Programmcode zu interpretieren und im Kopf auszuführen.

## HINWEISE ZUR DURCHFÜHRUNG

»Code Mainia« enthält 64 Karten verschiedener Farben. Jede Farbe repräsentiert ein Problemfeld und erweitert die fachliche Kompetenz.

### Farbkategorien

- › **Grün (24 Karten):** Zuweisung, einfache arithmetische Operationen (Addition, Subtraktion, Multiplikation)
- › **Gelb (18 Karten):** Inkrement/Dekrement ( $++$ ,  $--$ ), schwierigere Rechenoperationen (Modulo, Verknüpfung einfacher Rechenoperationen), Unterschied Zuweisung/Vergleich, weitere Variablen zur Irritierung
- › **Orange (12 Karten):** Verzweigungen, Vergleichsoperatoren ( $<=$ ,  $>=$ ,  $<$ ,  $>$ ,  $==$ ), logische Verknüpfung ( $\&\&$ )
- › **Rot (10 Karten):** While- und For-Schleifen
- › »Code Mainia« kann in zwei Varianten gespielt werden. Bei der didaktischen Variante *Grundspiel* wird nach jedem Ablegen einer Karte deren Bedeutung und Auswirkung auf den Variablenwert genannt. Durch das kontinuierliche Erklären und das häufige Wiederholen der Anweisungen kann das Wissen sehr gut vertieft werden. Die Lerngruppen können hier heterogen sein.

- › Die Variante *Dynamische Erweiterung* ist kompetitiv. Hier treten die Schüler in einen Wettstreit ein und versuchen, den Code so schnell wie möglich auszuwerten. Sobald der Spieler, der an der Reihe ist, den Programmcode nachvollzogen hat, kann er durch Ablegen seiner Karte die vorherige Spielkarte verdecken. Langsame Spieler können den aktuellen Variablenwert somit nicht mehr nennen. Da dies sehr schnell passieren kann, empfiehlt sich die Bildung leistungshomogener Gruppen.
- › Pro Gruppe von 3–5 SuS wird ein Spielset ausgegeben. Der Lehrende sagt die Farben der Themen an, die in dieser Stunde behandelt werden sollen. Die SuS sortieren alle andersfarbigen Karten aus und legen sie zurück in die Spielebox. Theoretisch kann jede Farbkategorie einzeln gespielt werden. Die grünen Karten sollten jedoch immer mit im Spiel sein, da sie für die Regulierung von Variablenwerten benötigt werden und auch als Grundstock dienen.
- › Je nachdem, mit welchen Kategorien gespielt wird, ändert sich die Anzahl der Spielkarten. Es können auch einzelne unerwünschte Karten aus dem Set entfernt werden.
- › Für das *Grundspiel* empfiehlt es sich, ca. 5–10 Karten als Nachziehstapel bereitzulegen. Bei der *Dynamischen Erweiterung* werden entweder alle Karten gleichmäßig unter den Spielern aufgeteilt oder der Lehrende gibt eine Kartenzahl vor. Da es bei der *Dynamischen Erweiterung* keinen Nachziehstapel gibt, werden nicht verwendete Karten beiseitegelegt.
- › Bei Einführung des Spiels sollte darauf geachtet werden, dass nicht alle Kategorien gleichzeitig verwendet werden. Insbesondere die roten Karten (Schleifen) sind selbst für erfahrene Programmierer recht komplex und könnten die SuS zunächst überfordern. Daher sollten sie erst eingeführt werden, wenn das *Grundspiel* mindestens einmal komplett durchgespielt wurde.
- › Bei der *Dynamischen Erweiterung* ist zu empfehlen, einen stärkeren Fokus auf den

edukativen Charakter zu legen, damit dieser in der Aufregung des Spiels nicht verloren geht. Vor allem ist darauf zu achten, dass eine falsche Lösung nicht nur als »falsch« deklariert wird, sondern der Ursprung des Fehlers gefunden wird.

## HINWEISE ZUM UNTERRICHTSVERLAUF

- › Aufgrund der unterschiedlichen Farbkategorien ist »Code Mainia« für viele Themenbereiche der praktischen Informatik im Unterricht einsetzbar. Beispielsweise kann nach einer kurzen Einführung das Thema Variablenkonzept unmittelbar spielerisch angewendet werden. Mit wachsendem Lernfortschritt können nach der Einführung von Verzweigungen oder Schleifen weitere Karten hinzugenommen werden, und das Spiel kann (erneut) gespielt werden.
- › »Code Mainia« eignet sich gut, um bereits erarbeiteten Unterrichtsstoff zu vertiefen,

Inhalte nach einer Lerneinheit zu wiederholen oder den Leistungsstand festzustellen.

- › Im Unterrichtsverlauf lässt sich das Spiel deshalb besonders gut nach jedem eingeführten Programmierkonzept (Farbkategorie), zur Übung und Festigung am Ende einer Lerneinheit oder nach einer längeren Pause (wie nach den Ferien) zur Auffrischung einsetzen.
- › Zudem kann das Spiel während der Lerneinheit »Programmierung« genutzt werden, um Wissen anzuwenden und zu festigen sowie das schnelle Lesen und Verstehen eines Programmcodes zu üben.

## BEISPIEL FÜR DEN KONKRETEN EINSATZ IM UNTERRICHT

**Klassensituation:** 15 SuS, Klassenstufe 10, Doppelstunde.

**Raumeigenschaften:** Tische können in Gruppen zusammengestellt werden. Das Spiel lässt sich ohne weitere Hilfsmittel überall spielen.

**Unterrichtseinheit:** Einstieg in die Programmierung.

**Vorwissen:** Variablenkonzept.

**Stundenverlauf:** In der ersten Stunde werden Verzweigungen eingeführt, in der zweiten wird »Code Mania« gespielt.

**Durchführung:** Die Spielanleitung wird ausgegeben. Die SuS erarbeiten sich selbstständig die Anleitung zum *Grundspiel* und beginnen mit dem Spiel. Das Spiel dauert inklusive Einarbeitung ca. 15–20 Minuten.

Nach mindestens einem erfolgreichen Spieldurchlauf können die SuS zur Spielvariante *Dynamische Erweiterung* wechseln. Dabei achtet der Lehrende darauf, dass der edukative Charakter nicht verloren geht – das Spielen der *Dynamischen Erweiterung* kann emotionsgeladen sein und für Unruhe sorgen.

**Abschluss:** Am Ende der Stunde sollten etwa 5 Minuten für das Aufräumen der Sets eingeplant werden, damit sie fürs nächste Mal wieder vollständig sind.

**Folgestunde:** Einführung des Schleifenkonzepts.