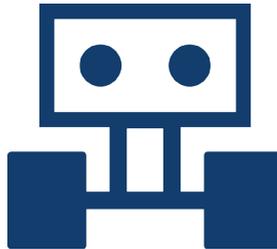


# ICDL Professional **ROBOTIK**

Basic



Lernzielkatalog

## Zweck dieses Dokuments

Dieses Dokument listet die Lerninhalte für das Modul *ICDL Robotik Basic* auf und beschreibt, welche Fertigkeiten von den Absolvent\*innen des Moduls erwartet werden. Die theoretischen und praktischen Aufgaben der Tests zu diesem Modul beruhen auf den Inhalten dieses Lernzielkatalogs. Approbierte Lernmaterialien decken dessen Inhalte ab.

Der ICDL ist eine Initiative der ICDL Foundation und wird in Österreich von der OCG betreut.

## ICDL Foundation

The Grange  
Stillorgan Road  
Blackrock  
Co. Dublin  
Republic of Ireland  
Web: [www.icdl.org](http://www.icdl.org)

## Österreichische Computer Gesellschaft (OCG)

Wollzeile 1  
A-1010 Wien  
Tel: +43 1 512 02 35-0  
E-Mail: [info@ocg.at](mailto:info@ocg.at)  
Web: [www.ocg.at](http://www.ocg.at)

## Hinweis

Die aktuelle deutschsprachige Version von ICDL Lernzielkatalogen für Österreich ist auf der ICDL Website [www.icdl.at](http://www.icdl.at) veröffentlicht.

## Haftung

Die OCG hat dieses Dokument mit Sorgfalt erstellt, kann aber weder Richtigkeit und Vollständigkeit der enthaltenen Informationen zusichern noch Haftung für durch diese Informationen verursachte Schäden übernehmen.

## Urheberrecht

© ICDL Foundation

# ROBOTIK BASIC

In diesem Modul werden die Grundprinzipien der Robotik, der Zusammenbau, die Programmierung und die Steuerung eines einfachen Roboters erläutert.

## LERNZIELE

Die Absolvent\*innen können

- Schlüsselkonzepte im Zusammenhang mit Robotern und Robotiksystemen verstehen und Beispiele für Roboter identifizieren,
- die wesentlichen Teile eines Roboters und ihre Funktion erkennen, einschließlich Mikrocontroller, Antriebe, Sensoren und Energiequellen,
- die Elemente eines einfachen Steuerungssystems verstehen und ein Steuerungssystem testen,
- grundlegende Programmierkonzepte verstehen sowie ein Programm in einer visuellen Programmiersprache erstellen und ausführen,
- einen Roboter einrichten, Bewegungen implementieren und den Roboter in einer Umgebung steuern.

## 1 GRUNDKONZEPTE DER ROBOTIK

### 1.1 Roboter und automatisierte Systeme

- 1.1.1 Definition der Begriffe Roboter und Robotiksysteme.
- 1.1.2 Verstehen, dass Roboter ferngesteuert, teilautonom oder autonom sein können.
- 1.1.3 Verstehen, dass Roboter stationär oder mobil sein können.

### 1.2 Verwendung von Robotern

- 1.2.1 Häufige Anwendungen von Robotern in verschiedenen Umgebungen kennen, wie: Zuhause, Schule, Produktion, Gesundheitswesen.
- 1.2.2 Fortgeschrittene Anwendungen von Robotern kennen, wie: selbstfahrende Autos, robotergestützte Operationen.
- 1.2.3 Ethische Probleme bei der Verwendung von Robotern kennen, wie: Menschen Schaden zufügen.

## 2 ROBOTIK TEILE

### 2.1 Wesentliche Teile und Komponenten

- 2.1.1 Wesentliche Teile von Robotern erkennen, wie: Antrieb, Mikrocontroller, Sensoren, Energiequellen.
- 2.1.2 Komponenten eines Roboter-Sets kennen, wie: Chassis, Elektronikteile, Kabel, Werkzeuge und Teile für den Zusammenbau.

### 2.2 Mikrocontroller

- 2.2.1 Wissen, dass der Mikrocontroller Informationen von Eingabegeräten wie Sensoren sammelt, Programme ausführt und Ausgabegeräte wie LEDs und Audiogeräte steuert.
- 2.2.2 Wesentliche Anschlüsse von Mikrocontrollern kennen, wie: Stromanschluss, USB, kabellos, Ein- und Ausgang.

### 2.3 Antriebsysteme

- 2.3.1 Hauptteile von Aktuatoren kennen, wie: Schalter und Motor
- 2.3.2 Verstehen, dass der Aktuator elektrische Energie in mechanische Energie umwandelt, um den Roboter anzutreiben.

### 2.4 Sensoren

- 2.4.1 Verstehen, dass ein Sensor Änderungen in der Umgebung wahrnehmen kann, wie: Lichtstärke, Entfernung, Winkel.
- 2.4.2 Die Funktion von verschiedenen Sensortypen kennen, wie: Licht, Klang, Gyroskop.

### 2.5 Fortbewegung und Energiequellen

- 2.5.1 Teile kennen, welche die Bewegung eines Roboters unterstützen, wie: Arme, Räder.
- 2.5.2 Energiequellen kennen, wie: Batterien, Sonnenenergie.

## 3 EINFACHE STEUERUNGSSYSTEME

### 3.1 Übersicht der Steuerungssysteme

- 3.1.1 Elemente eines Steuerungssystems kennen; grundlegende Arten einer Steuerung verstehen, wie: offener und geschlossener Regelkreis.
- 3.1.2 Anbindungen an den Mikrocontroller verstehen, wie: Taster, Strom, Motor, USB-Eingang, kabellose Technologien, Sensoren, Ausgabegeräte.
- 3.1.3 Verbindungen zum Mikrocontroller in einem Blockdiagramm verstehen.
- 3.1.4 Ein einfaches Steuerungssystem aufsetzen, z. B mit Energiequellen, Motoren und Sensoren.

### 3.2 Einfache Steuersysteme testen

- 3.2.1 Vordefinierte Programme ausführen, um Ausgabewerte bereitzustellen, wie: Lichtstärke, Klang, Entfernung, Winkel.
- 3.2.2 Verstehen, dass es zwischen Eingang und Ausgabe der Daten eine Verzögerung gibt.
- 3.2.3 Verstehen, dass das Verändern von Variablen in einem Programm die Ausgabe beeinflusst.

## 4 VISUELLE PROGRAMMIERUNG

### 4.1 Programmiergrundlagen

- 4.1.1 Definition der Begriffe Programm und Programmiersprache.
- 4.1.2 Wissen, dass Blöcke grundlegende Elemente in einer visuellen Programmiersprache sind; häufige Blockkategorien kennen, wie: Ereignisse, Steuerung.
- 4.1.3 Typische Tätigkeiten bei der Erstellung eines Programms kennen, wie: Analysieren einer Aufgabe, Entwerfen einer Lösung, Schreiben eines Programms, Testen und Verbessern eines Programms.
- 4.1.4 Grundlegende Elemente eines Programms kennen, wie: Ablauf, Entscheidungen, Schleifen.
- 4.1.5 Verstehen wie ein Flussdiagramm verwendet werden kann, um die Schritte einer Lösung darzustellen.

## 4.2 Konstante, Variable

- 4.2.1 Zwischen den Begriffen Konstante und Variable im Kontext eines Programms unterscheiden.
- 4.2.2 Neue Variablen erstellen und passende Werte zuweisen.

## 4.3 Ereignisse, Steuerung

- 4.3.1 Verwendung eines Ereignisblocks in einem Programm, wie: wenn.
- 4.3.2 Verwendung eines Steuerungsblocks in einem Programm, wie: warten, warten bis.
- 4.3.3 Eine Schleife oder endlose Fortsetzung mit Blöcken implementieren, wie: für immer, wiederholen.
- 4.3.4 Bedingungen mit Blöcken implementieren, wie: wenn, dann, sonst.
- 4.3.5 Logische Operatoren verwenden, wie: und, nicht, oder.

## 4.4 Erstellen und Ausführen eines Programms

- 4.4.1 Einen Plan skizzieren und ein Problem lösen, wie: Steuerung einer Ausgabe, eine Reihe von Aktionen durchführen.
- 4.4.2 Zeichnen eines Flussdiagramms, um die Schritte einer Lösung abzubilden.
- 4.4.3 Erstellen eines Programms in einer visuellen Programmiersprache, um ein Problem zu lösen, wie: Steuerung einer Ausgabe, eine Reihe von Aktionen durchführen.
- 4.4.4 Verstehen, dass es mehr als einen Weg gibt ein Programm zu schreiben, um dasselbe Problem zu lösen.
- 4.4.5 Ausführen eines Programms; Identifizieren und Lösen von Fehlern in einem Programm.

# 5 ARBEITEN MIT ROBOTERN

## 5.1 Einrichten

- 5.1.1 Sicherheitsrichtlinien verstehen und implementieren, wie: sicherer Umgang mit elektronischen Teilen und Werkzeug; Bewusstsein um die eigene Sicherheit und um die anderer.
- 5.1.2 Zusammenbauen eines Roboters mit dem vorhandenen Werkzeug.

## 5.2 Implementierung der Roboterbewegung

- 5.2.1 Implementierung von einfachen Roboterbewegungen, wie: Stopp, Vorwärts- und Rückwärtsbewegung, Drehen.
- 5.2.2 Verständnis für die Zusammenhänge zwischen Energie, Entfernung, Geschwindigkeit und Zeit in der Roboterbewegung.
- 5.2.3 Anwenden von Konzepten wie Energie, Entfernung, Geschwindigkeit und Zeit, um Bewegungen zu steuern: Vorwärts- und Rückwärtsbewegung; verstehen, dass Schwung und Reibung die Bewegung beeinflussen können.
- 5.2.4 Zusammenhänge von Energie, Rotationsgeschwindigkeit und Winkel der Rotation in der Roboterbewegung verstehen.

## 5.3 Implementierung der Robotersteuerung

- 5.3.1 Verwendung eines Roboters, um Sensordaten zu sammeln, wie: Entfernung, Klang, Winkel, Licht
- 5.3.2 Bauen, Testen und Verbessern eines Programms, um den Roboter mittels eines Sensors zu steuern, wie: Licht, Klang, Gyroskop.
- 5.3.3 Die Wichtigkeit des Testens verstehen, um Fehler zu beseitigen
- 5.3.4 Verstehen, dass manche Fehler zufällig auftreten, wie: Staub, unbekannte Variablen.

## 5.4 Steuerung in einer Umgebung

- 5.4.1 Navigation eines Roboters in einer Umgebung, um Aufgaben mithilfe verschiedener Funktionalitäten abzuschließen, wie: einer Linie folgen oder ausweichen; einem Objekt / einem Hindernis folgen oder ausweichen; eine Rampe hinauf- oder hinunterfahren
- 5.4.2 Navigation eines Roboters in einer Umgebung, um verschiedene Szenarien mithilfe passender Kombinationen von Bewegung und Funktionalitäten abzuschließen
- 5.4.3 Wissen, dass Teamwork bei der gemeinsamen Arbeit an einem Roboter wichtig ist; Bedeutung von Fähigkeiten/Skills kennen, wie: Planung, Kommunikation, Aufgabenzuteilung

# ICDL MODULE IM ÜBERBLICK

## GRUNDLAGEN



Computer-Grundlagen



Online-Grundlagen

## OFFICE ANWENDUNGEN



Tabellenkalkulation



Textverarbeitung



Präsentation

## GOOD PRACTICE



Datenbanken anwenden



IT-Security



Online-Zusammenarbeit



Bildbearbeitung



Computing



Webediting

## KI UND ROBOTIK



Robotik



Künstliche Intelligenz

## ADVANCED



Tabellenkalkulation  
Advanced



Textverarbeitung  
Advanced



Präsentation  
Advanced



Datenbank Advanced

