



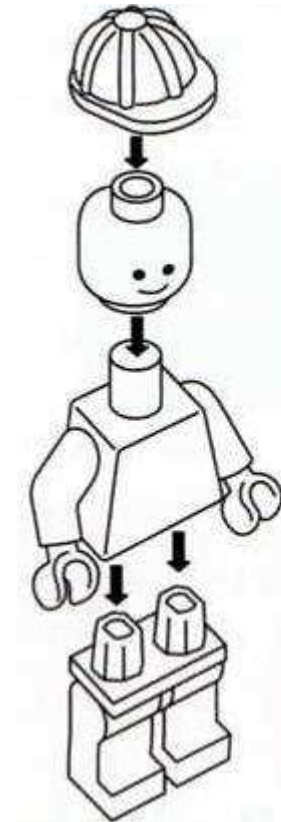
# Informatikunterricht in der SEK II

Voraussetzungen - Inhalte - Prüfungen



## Du kannst

- logisch Denken
- Selbstständig arbeiten
- Mit anderen kooperieren
- Abstrakte Konzepte nachvollziehen
- Lösungen erläutern



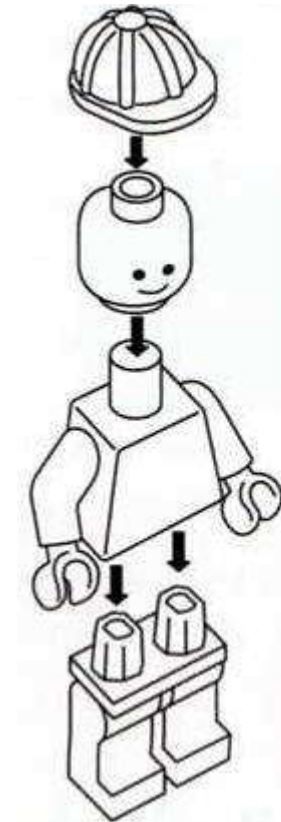
"Dieses Foto" von Unbekannter Autor ist lizenziert gemäß [CC BY-NC](#)

## Du kannst

- logisch Denken
- Selbstständig arbeiten
- Mit anderen kooperieren
- Abstrakte Konzepte nachvollziehen
- Lösungen erläutern



Und du hast Spaß daran?



"Dieses Foto" von Unbekannter Autor ist lizenziert gemäß [CC BY-NC](#)

## Du willst lernen

- wie man programmiert
- wie man strukturiert Probleme löst
- welche Auswirkungen Informatiksysteme auf unsere Gesellschaft haben
- wie man große Datenmengen verwaltet
- was der Aufbau deines PCs ist
- wie man große Projekte plant
- wie das Internet funktioniert
- was Menschen besser können werden als Roboter



"Dieses Foto" von Unbekannter Autor ist lizenziert gemäß [CC BY-SA](#)



## Das brauchst du NICHT

- Du brauchst keine Vorkenntnisse in der Programmierung
- Du muss Informatik nicht in der Sek I belegt haben.



"Dieses Foto" von Unbekannter Autor ist lizenziert gemäß [CC BY-SA-NC](#)

## Informatiksysteme

- Einführung in die Nutzung von Informatiksystemen und in grundlegende Begrifflichkeit
- Darstellung ganzer Zahlen und Zeichen in Binär- und Hexadezimalcode.

## OO-Modellierung und Programmierung (JAVA)

- Einführung des Klassen- und Objektbegriffe
- Regeln der strukturierten Programmierung und deren Dokumentation
- Implementation, Ausführen und Testen einfacher Programme

## Such- und Sortieralgorithmen

- Analyse einfacher Suchalgorithmen am Beispiel der linearen und binären Suche
- Analyse einfacher Sortieralgorithmen am Beispiel Insertion- und Selection-Sort
- Entwurf eines eigenen Sortieralgorithmus in Pseudocode

## Geschichte der EDV / Datenschutz-Grundlagen

- Geschichte und Entwicklung von Informatiksystemen
- Bewertung der Auswirkungen des Einsatzes von Informatiksystemen anhand von Fallbeispielen

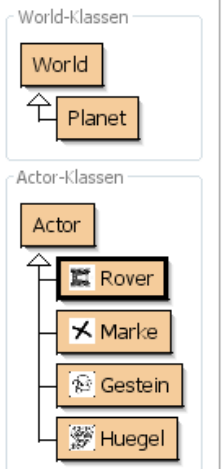
## OO-Modellierung und Programmierung (JAVA)

- Einführung des Klassen- und Objektbegriffe
- Regeln der strukturierten Programmierung und deren Dokumentation
- Implementation, Ausführen und Testen einfacher Programme

Ich bin bereit



Exportieren...



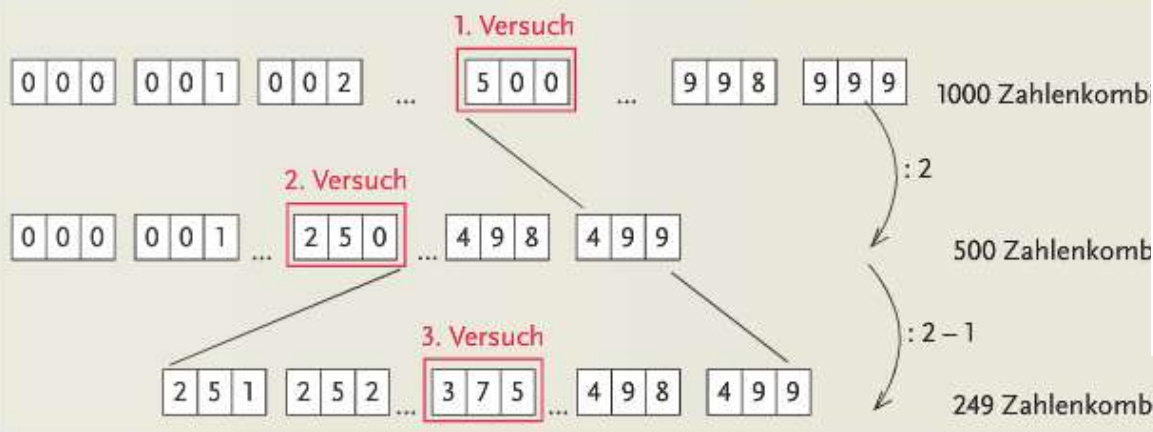
```
/**
 * Der Rover soll einen unregelmäßiges Hügelfeld umfahren. Das Ende der Umfahrung ist durch eine Marke gekennzeichnet
 */
public void umfahre_huegel_3()
{
    //Wiederhole Anweisungsblock so lange, bis Rover auf einem Feld mit Marke steht
    while(!markeVorhanden())
    {
        // Solange rechts ein Hügel ist, fährt : Rover fährt geradeaus weiter
        while(huegelVorhanden("rechts"))
        {
            while(huegelVorhanden("vorne"))
            {
                drehe("links");
            }
            fahre();
        }
        // Rover dreht rechts und fährt ein Feld
        drehe("rechts");
        fahre();
    }
}
```

## Such- und Sortialgorithmen

- Analyse einfacher Suchalgorithmen am Beispiel der linearen und binären Suche
- Analyse einfacher Sortialgorithmen am Beispiel Insertion- und Selektion-Sort
- Entwurf eines eigenen Sortialgorithmus in Pseudocode

## Suchen in sortierten Daten – Binäre Suche

Paul bittet seinen Bruder Lukas, ihm sein Fahrrad zu leihen. Lukas zögert: „Du kannst es haben, wenn du die Zahlenkombination des Fahrrad-schlusses errätst! Ich gebe dir zehn Rateversuche und sage dir außerdem jedes Mal, ob dein Versuch zu groß oder zu klein geraten ist.“ Lukas hat ein dreistelliges Zahlenschloss für sein Fahrrad. Es kommen also insgesamt 1000 verschiedene Kombinationen von 000 bis 999 in Frage. Hat Paul mit seinen zehn Versuchen eine faire Chance, das Fahrrad zu bekommen?



```
boolean binaereSuche (int [] a, int suchschluessel)
int links, rechts, mitte
links = 0
rechts = größter Index des Arrays a
solange (links <= rechts)
wiederhole
    mitte = Index zwischen rechts und links //ggf. auf
    wenn (a[mitte] == suchschluessel) dann
        return true
    ende wenn
    wenn (a[mitte] < suchschluessel) dann
        links = mitte + 1
    ende wenn
    wenn (a[mitte] > suchschluessel) dann
        rechts = mitte - 1
    ende wenn
ende wiederhole
return false
```

Die binäre Suche schließt bei jedem Schritt etwa die Hälfte aller möglichen Objekte aus.



# Geschichte der EDV / Datenschutz-Grundlagen

- Geschichte und Entwicklung von Informatiksystemen
- Bewertung der Auswirkungen des Einsatzes von Informatiksystemen anhand von Fallbeispielen



Alle Fotos von  
Unbekannter  
Autor [CC BY-  
SA](#)

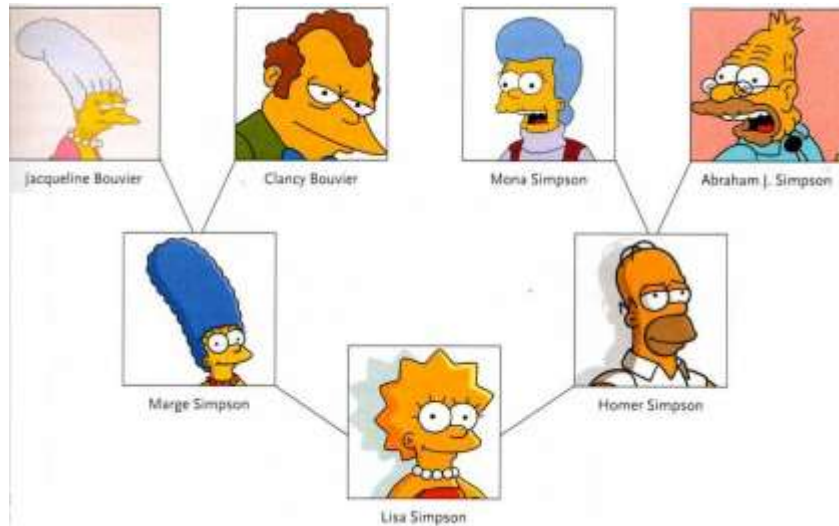


"Dieses Foto" von Unbekannter Autor ist lizenziert gemäß [CC BY-NC-ND](#)

# Ausblick Inhalte der Q1 und Q2

Unbekannter Autor ist lizenziert gemäß [CC BY-SA-NC](https://creativecommons.org/licenses/by-sa/4.0/)

## Informatik, Mensch und Gesellschaft



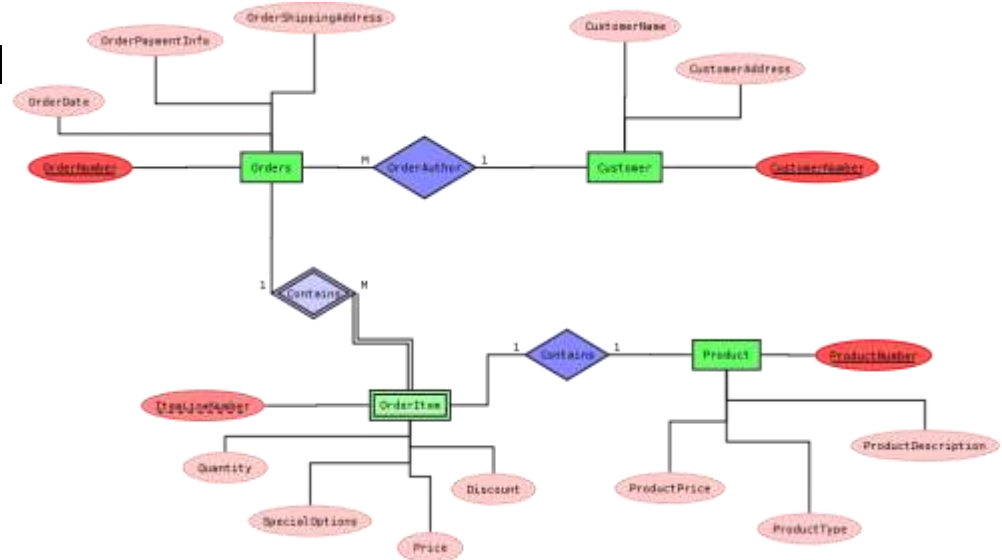
## Vertiefung Praktische Informatik

- Objektorientierung
- Datenstrukturen (Liste, Bäume,...)
- Komplexe Sortieralgorithmen

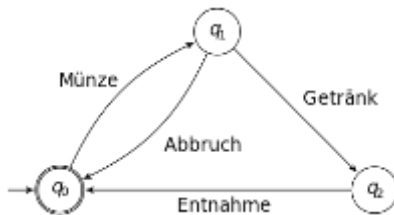
# Ausblick Inhalte der Q1 und Q2

## Datenbanksysteme

- Darstellung im ER-Modell
- Relationale Datenbank
- SQL



"Dieses Foto" von Unbekannter Autor ist lizenziert gemäß [CC BY-SA](#)



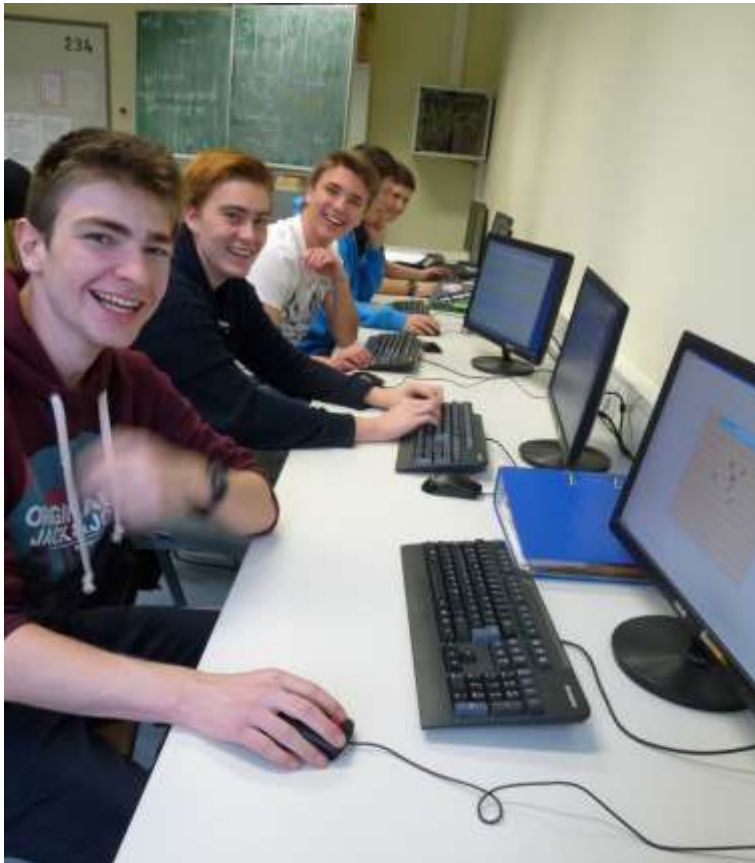
"Dieses Foto" von Unbekannter Autor ist lizenziert gemäß [CC BY-SA](#)

## Theoretische Informatik

- Wie funktioniert ein Getränkeautomat?
- Reguläre Sprache und endliche Automaten

- zählt zum naturwissenschaftlichen Bereich (als 2. Fach)
- in der EPH eine Klausur pro Halbjahr
- in der Qualifizierungsphase zwei Klausuren pro Halbjahr
- als drittes oder viertes Abiturfach wählbar
- Leistungskurs wird angeboten

# Auf ein Wiedersehen!



*Wir freuen uns  
auf dich!*

*Deine  
Informatikfachschaft*