

# Schulinterner Lehrplan

## Informatik

### Gymnasium Essen Nord-Ost (GENO)

Sekundarstufe I & II

Vorläufige Fassung (Stand: August 2025)

#### 1. Einleitung

Dieser schulinterne Lehrplan (SILP) für das Fach Informatik am GENO basiert auf:

- den Vorgaben des Kernlehrplans Sek I (Gymnasium G9, ab 2019),
- den Abiturvorgaben Informatik NRW für die gymnasiale Oberstufe (GK),
- dem Medienkompetenzrahmen NRW,
- sowie dem Schulgesetz NRW (§29, §70).

Er ist als **vorläufige Version** zu verstehen, da neue Kernlehrpläne Sek II in naher Zukunft erwartet werden. Anpassungen erfolgen nach Veröffentlichung.

#### 2. Fachliche Leitlinien am GENO

Das Fach Informatik am GENO vermittelt Kompetenzen zur aktiven, reflektierten und kreativen Gestaltung der digitalen Welt. Es verbindet algorithmisches Denken, Softwareentwicklung, Datenverarbeitung und gesellschaftliche Reflexion.

Ein zentrales Arbeitsmedium ist das kostenlose Online-Schulbuch [inf-schule.de](https://inf-schule.de).

##### Besonderheiten des GENO:

- Frühe Einführung in Klasse 6 (2 Stunden/Woche),
- Differenzierungskurse in Jg. 9/10 (3 Stunden/Woche),
- Oberstufe als Grundkurs (3 Stunden/Woche),
- Integration von **3D-Druckern** und einem programmierbaren **Roboterarm (Dobot Magician Lite)** in den Unterricht.

#### 3. Jahrgangsstufe 6 (Pflichtfach, 2 Wochenstunden)

| Unterrichtsvorhaben     | Kompetenzen (KLP)                            | Methoden/Materialien                                                        |
|-------------------------|----------------------------------------------|-----------------------------------------------------------------------------|
| Digitale Welt entdecken | Informatiksysteme verstehen, Medienkompetenz | Tablets/PCs, <a href="https://inf-schule.de">inf-schule.de</a> („Einstieg“) |

| Unterrichtsvorhaben                 | Kompetenzen (KLP)                      | Methoden/Materialien                                      |
|-------------------------------------|----------------------------------------|-----------------------------------------------------------|
| Sicher im Netz                      | Kommunizieren, Reflektieren            | Passwort-Übungen, Datenschutz-Quiz                        |
| Algorithmen & Programmieren I       | Algorithmen entwickeln, Modellieren    | Scratch/Blockly, Calliope mini                            |
| Daten darstellen                    | Daten & Codierungen                    | Pixelgrafik, ASCII-Übungen                                |
| Projektarbeit: Informatik im Alltag | Kooperieren, Produzieren, Präsentieren | Gruppenprojekte, Präsentationen, 3D-Druck kleiner Modelle |

#### 4. Differenzierungskurse Jahrgang 9/10 (3 Wochenstunden)

| Unterrichtsvorhaben                 | Kompetenzen (KLP)                 | Kreative Ideen/Methoden                                    |
|-------------------------------------|-----------------------------------|------------------------------------------------------------|
| Einführung in Programmierung        | Algorithmen, Daten, Strukturieren | Turtle-Grafik, Graphische Programmierung                   |
| Objektorientierung (Grundlagen)     | OOP verstehen, Klassen/Objekte    | Programmierung einfacher Spiele                            |
| Web & Daten                         | Datenbanken, Informationssysteme  | Netzwerktechnik, sozialer Umgang im Internet, Rechtsrahmen |
| Roboter & Sensoren                  | Problemlösen, Modellieren         | LEGO Mindstorms/Calliope, Dobot Magician Lite (Roboterarm) |
| Projektarbeit (App/Spiel/3D-Modell) | Produzieren, Präsentieren         | Teamprojekt mit Präsentation, Einsatz von 3D-Druckern      |

#### 5. Gymnasiale Oberstufe (EF, Q1, Q2, GK, 3 Wochenstunden)

**ACHTUNG: KLP NOCH NICHT VERÖFFENTLICH!**

| Jahrgang | Unterrichtsvorhaben                                                                        | Besondere Schwerpunkte                                                            |
|----------|--------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|
| EF (11)  | Programmierung mit Java (Datentypen, Anweisungen, Syntax, Sortiervorgang außer Merge Sort) | Grundlagen, Debugging, Testen                                                     |
| Q1 (12)  | Rekursion, Merge Sort, Quicksort, Automaten & Grammatiken, Netzwerke                       | Abiturvorgaben NRW, Modellierung von Prozessen, ggf. Robotik-Erweiterungen        |
| Q2 (13)  | Datenbanken, Algorithmen in Binärbäumen, Suchen & Sortieren                                | SQL, Normalisierung, Abiturvorbereitung, projektorientiert mit 3D-Drucker/Robotik |

## 6. Grundsätze der Leistungsbewertung

- Gewichtung: ca. 50 % praktische/mündliche Leistungen (Programmieren, Projekte, Präsentationen), 50 % schriftliche Arbeiten (Tests, Klausuren in der Oberstufe).
- In Differenzierungskursen und Projektphasen können praktische Arbeiten mit dem **3D-Drucker** oder dem **Roboterarm** in die Bewertung einfließen.

## 7. Umsetzung des Medienkompetenzrahmens NRW

Alle sechs Kompetenzbereiche (Bedienen & Anwenden, Informieren & Recherchieren, Kommunizieren & Kooperieren, Produzieren & Präsentieren, Analysieren & Reflektieren, Problemlösen & Modellieren) werden umgesetzt.

Der Einsatz von **3D-Druckern** und **Robotik-Projekten** stärkt insbesondere:

- Produzieren & Präsentieren,
- Problemlösen & Modellieren.

## 8. Qualitätssicherung & Evaluation

Der SILP wird jährlich in der Fachkonferenz Informatik evaluiert und angepasst. Besondere Schwerpunkte:

- Integration neuer KLP-Vorgaben,
- Umsetzung des Medienkonzepts,
- Erfahrungen aus Projekten mit 3D-Druck und Robotik.

## 9. Hinweise & Kontakt

Dieser Lehrplan ist eine **vorläufige Fassung**, da neue Kernlehrpläne für die Sek II erwartet werden.

Fragen und Rückmeldungen bitte an:

**Kemper@gtgeno.de**

## 10. Quellen & Links

- Ministerium für Schule und Bildung NRW: Lehrplannavigator Informatik
- Abiturvorgaben Informatik NRW
- Medienkompetenzrahmen NRW
- Amtliche Schulmail/BASS
- Online-Schulbuch: [inf-schule.de](http://inf-schule.de)

## 11. Details einzelner Unterrichtsvorgaben

### **Digitale Welt entdecken**

In diesem Unterrichtsvorhaben lernen die Schüler\*innen die grundlegenden Komponenten eines Computers, einfache Netzwerke und digitale Werkzeuge kennen. Sie üben sich in der Orientierung in der digitalen Welt, erkennen die Vor- und Nachteile digitaler Medien und entwickeln erste Medienkompetenz.

Geförderte Kompetenzen:

- Technisches Verständnis von Informatiksystemen
- Bedienen und Anwenden von Hard- und Software
- Reflexion über digitale Mediennutzung

Unterstützung für die SchülerInnen:

Durch praxisnahe Übungen am PC oder Tablet werden Neugier und logisches Denken gefördert. Dieses Vorhaben legt die Grundlage für spätere Programmieraufgaben und das Arbeiten mit Robotik und 3D-Druckprojekten.

### **Sicher im Netz**

Die Schüler\*innen setzen sich aktiv mit Themen wie Passwortsicherheit, Datenschutz, Privatsphäre und Cybermobbing auseinander. Anhand kleiner Übungen und Quizen erlernen sie, wie sie sich sicher in digitalen Räumen bewegen.

Geförderte Kompetenzen:

- Informieren und Recherchieren im Netz
- Kommunizieren und Kooperieren in digitalen Kontexten
- Reflexion der eigenen Mediennutzung

Unterstützung für die SchülerInnen:

Dieses Vorhaben sensibilisiert die Schüler\*innen früh für die Risiken und Chancen digitaler Medien, was für die Arbeit mit vernetzten Systemen (z.B. Roboterarm oder 3D-Druck) unerlässlich ist.

### **Algorithmen & Programmieren I**

Die Schüler\*innen entwickeln erste eigene Algorithmen und setzen diese mit visuellen Programmiersprachen wie Scratch oder Blockly um. Sie lernen einfache Schleifen, Bedingungen und Variablen kennen.

Geförderte Kompetenzen:

- Logisches und algorithmisches Denken
- Strukturieren von Problemstellungen
- Produzieren und Präsentieren eigener Programme

Unterstützung für die SchülerInnen:

Diese Basisbefähigung bereitet auf die späteren Java-Kurse vor und ermöglicht den

Schüler\*innen, ihre Kreativität in Projekten wie dem Programmieren des Dobot-Roboterarms umzusetzen.

### **Daten darstellen**

Die Schüler\*innen lernen, wie Daten kodiert und visualisiert werden können. Sie erstellen einfache Grafiken, Diagramme und Pixelbilder und verstehen die Grundlagen von Binärsystemen und Codierungen.

Geförderte Kompetenzen:

- Analysieren und Modellieren von Daten
- Problemlösen durch algorithmisches Denken
- Verknüpfung von Theorie und praktischer Umsetzung

Unterstützung für die SchülerInnen:

Die Schüler\*innen entwickeln ein Grundverständnis für Datendarstellung, das sie für die Programmierung und Datenanalyse in höheren Jahrgangsstufen benötigen.

### **Projektarbeit: Informatik im Alltag**

Die Schüler\*innen wählen kleine Projekte, bei denen sie Informatik in ihrem Alltag anwenden. Beispiele: kleine Spiele programmieren, einfache interaktive Geschichten oder erste 3D-Objekte erstellen.

Geförderte Kompetenzen:

- Kooperieren, Produzieren, Präsentieren
- Selbstständiges Planen und Umsetzen von Projekten
- Kreatives und problemlösendes Denken

Unterstützung für die SchülerInnen:

Die Arbeit mit 3D-Drucker und Roboterarm motiviert die Schüler\*innen, Theorie in greifbare Ergebnisse umzusetzen, fördert Teamarbeit und macht informatische Inhalte anschaulich.

## **4. Differenzierungskurse Jahrgang 9/10 (3 Wochenstunden)**

Einführung in Programmierung

Die Schüler\*innen vertiefen ihre Kenntnisse in Python oder Java. Sie entwickeln komplexere Programme, lernen Funktionen kennen und führen strukturierte Tests durch.

Geförderte Kompetenzen:

- Algorithmisches Denken und Strukturierung
- Systematisches Testen und Debugging
- Problemlösen und Abstrahieren

Unterstützung für die SchülerInnen:

Dieses Vorhaben bereitet die Schüler\*innen auf Oberstufenkurse vor und fördert gleichzeitig Selbstständigkeit und Durchhaltevermögen bei komplexen Aufgaben.

### **Objektorientierung (Grundlagen)**

Einführung in Klassen, Objekte, Methoden und Attribute. Schüler\*innen erstellen kleine Programme, die reale Objekte modellieren (z.B. ein Robotersystem).

Geförderte Kompetenzen:

- Modellieren von realen Situationen in der Informatik
- Strukturierte Problemlösung und Abstraktion
- Teamarbeit bei größeren Projekten

Unterstützung für die SchülerInnen:

Die Inhalte fördern logisches Denken und bereiten direkt auf die Programmierung von Robotik-Anwendungen vor (z.B. Dobot-Roboterarm).

### **Web & Daten**

Die Schüler\*innen lernen, Datenbanken zu nutzen, einfache Webanwendungen zu erstellen und Daten strukturiert zu speichern. SQL wird eingeführt.

Geförderte Kompetenzen:

- Datenbankgrundlagen und Informationssysteme verstehen
- Problemlösen und Modellieren realer Daten
- Projektorientiertes Arbeiten

Unterstützung für die SchülerInnen:

Verknüpfung von Informatik mit realen Anwendungen, z. B. Web-Apps oder interaktiven Dashboards für 3D-Druck-Projekte.

### **Roboter & Sensoren**

Schüler\*innen programmieren LEGO Mindstorms, Calliope und den Dobot-Roboterarm. Sie entwickeln Steuerprogramme, Sensorabfragen und automatisierte Abläufe.

Geförderte Kompetenzen:

- Problemlösen und Modellieren komplexer Systeme
- Anwendung von Programmierkenntnissen in realen Szenarien
- Teamarbeit und projektorientiertes Lernen

Unterstützung für die SchülerInnen:

Durch praxisnahe Anwendungen werden abstrakte Konzepte greifbar. Die Schüler\*innen sehen direkten Erfolg beim Programmieren von Robotikprojekten.

## **Projektarbeit (App/Spiel/3D-Modell)**

Abschlussprojekte: Schüler\*innen entwickeln eigene Anwendungen oder 3D-Modelle, steuern den Roboterarm oder erstellen interaktive Programme. Präsentation vor der Klasse.

Geförderte Kompetenzen:

- Eigenständige Planung und Durchführung eines Projekts
- Präsentation und Reflexion von Arbeitsergebnissen
- Kreatives und problemlösendes Denken

Unterstützung für die SchülerInnen:

Fördert Kreativität, Projektmanagementfähigkeiten und technische Fertigkeiten, kombiniert Informatik, Robotik und 3D-Druck.

## **EF (11. Jahrgang) – Programmierung mit Java (Datentypen, Anweisungen, Syntax, Sortierverfahren außer Merge Sort)**

In der Einführungsphase lernen die Schüler\*innen die Grundlagen der Java-Programmierung. Schwerpunkte sind Datentypen, Operatoren, Kontrollstrukturen, Methoden und einfache Sortieralgorithmen (außer Merge Sort). Sie entwickeln kleine Programme, debuggen Fehler und testen ihre Anwendungen systematisch.

Geförderte Kompetenzen:

- Algorithmisches Denken und Problemlösungskompetenz
- Strukturierte Analyse und Umsetzung von Aufgabenstellungen
- Selbstständige Planung, Durchführung und Überprüfung von Programmen

Unterstützung für die Schüler\*innen:

Die Schüler\*innen erwerben eine solide Grundlage für komplexere Programmierprojekte, die später in Q1/Q2 und in Differenzierungskursen umgesetzt werden. Erste Projekte können mit Roboterarm-Simulationen oder kleinen 3D-Druck-Projekten gekoppelt werden, um abstrakte Programmierkonzepte anschaulich zu machen.

## **Q1 (12. Jahrgang) – Rekursion, Merge Sort, Quicksort, Automaten & Grammatiken, Netzwerke**

Die Schüler\*innen vertiefen ihre Kenntnisse in fortgeschrittener Programmierung, indem sie rekursive Methoden, effiziente Sortieralgorithmen (Merge Sort, Quicksort) und Datenstrukturen anwenden. Sie lernen die Theorie von Automaten, Grammatiken und Netzwerken kennen und setzen sie in praktischen Aufgaben um.

Geförderte Kompetenzen:

- Abstraktes und logisches Denken durch rekursive Algorithmen
- Analyse und Optimierung komplexer Prozesse
- Modellierung realer Systeme (z. B. Netzwerk- oder Kommunikationsstrukturen)

Unterstützung für die Schüler\*innen:

Die Schüler\*innen entwickeln ein tieferes Verständnis von Effizienz und Strukturierung von Algorithmen, das sich auf praktische Aufgaben wie die Programmierung des Dobot-Roboterarms

oder Steuerung von 3D-Druck-Prozessen übertragen lässt. Außerdem fördert dieses Vorhaben analytische und problemlösende Fähigkeiten, die für das Abitur relevant sind.

## **Q2 (13. Jahrgang) – Datenbanken, Algorithmen in Binärbäumen, Suchen & Sortieren**

Die Oberstufe schließt mit der Anwendung von Datenbanken, fortgeschrittenen Algorithmen zur Suche und Sortierung in binären Bäumen und komplexen Projektarbeiten ab. Schüler\*innen erstellen Datenbankabfragen, normalisieren Tabellen und implementieren Suchalgorithmen. Geförderte Kompetenzen:

- Analysieren und Strukturieren komplexer Daten
- Entwerfen, Testen und Optimieren von Algorithmen
- Eigenständige Planung und Umsetzung von Projekten im Team

Unterstützung für die Schüler\*innen:

Schüler\*innen wenden ihre Kenntnisse praktisch an, z. B. durch die Erstellung eines interaktiven 3D-Modells oder die Steuerung des Dobot-Roboterarms über eine Datenbankanbindung. Projektorientierte Aufgaben fördern Teamarbeit, Präsentationskompetenz und die Verbindung von Theorie und Praxis.

## **Projektorientierte Oberstufenarbeit**

In der Oberstufe werden die theoretischen Inhalte regelmäßig in praxisnahe Projekte integriert. Beispiele:

- Programmierung eines autonomen Roboterarms, der Aufgaben in einer Simulation oder real umsetzt
- Entwicklung eines kleinen Datenbank-gestützten Systems, das mit 3D-Druck- oder Robotikkomponenten verbunden ist
- Abstrakte Algorithmusaufgaben visualisieren, z. B. Such- oder Sortierprozesse in binären Bäumen

Geförderte Kompetenzen:

- Eigenständige Planung und Umsetzung komplexer Projekte
- Interdisziplinäres Denken (Informatik, Technik, Design)
- Präsentation, Reflexion und Dokumentation der Arbeitsergebnisse

Unterstützung für die Schüler\*innen:

Die praxisorientierten Projekte ermöglichen den direkten Transfer der theoretischen Inhalte in die reale Welt, stärken Motivation, Problemlösekompetenz und Kreativität. Gleichzeitig werden die Schüler\*innen optimal auf die Abituranforderungen vorbereitet.