**HOGESCHOOL ROTTERDAM / CMI**

**Algoritmiek**

**INFDEV01-6A**

|  |
| --- |
| Aantal studiepunten: 3  Modulebeheerder: Youri Tjang |

# Modulebeschrijving

|  |  |
| --- | --- |
| **Modulenaam:** | Algoritmiek |
| **Modulecode:** | **INFDEV01-6A** |
| **Aantal studiepunten en studiebelastinguren:** | Dit studieonderdeel levert 4 studiepunten op, hetgeen overeenkomt met een studielast van 112 uur.  De verdeling van deze 112 uur over de collegeweken is als volgt:  Begeleide colleges:  - gedurende 8 weken: 8 \* 200 minuten 25 uur  Onbegeleide uren per week:  - bestuderen theorie (8 \* 3 uur per week) 24 uur  - uitwerken van opdrachten (8 \* 5 uur per week) 40 uur  Totaal 84 uur |
| **Vereiste voorkennis:** | DEV01-4 |
| **Werkvorm:** | Werkcollege (theorie) en practicum |
| **Toetsing:** | Theorietentamen + praktische opdrachten |
| **Leermiddelen:** | * Sedgewick, R., & Wayne, K. (2011). *Algorithms* (4th ed.). Addison-Wesley Professional * Online lesmateriaal |
| **Draagt bij aan competentie:** | Realiseren |
| **Leerdoelen:** | * Je bent bekend met concepten van datastructuren en algoritmen * Je kan fundamentele datastructuren en algoritmes zelf implementeren * Je kan de efficiency van algoritmen (tijd en resource) analyseren * Je kan problemen herkennen en daar het juiste algoritme voor implementeren. |
| **Inhoud:** | Fundamentele datastructuren, Collection datastructuren, performance en  toepassingen. |
| **Opmerkingen:** |  |
| **Modulebeheerder:** | Youri Tjang |
| **Datum:** | 13 juli 2015 |

# 1. Algemene omschrijving

## Inleiding

In deze module maak je je vaardig in het domein van algoritmes en datastructuren.

## Relatie met andere onderwijseenheden

In deze onderwijsperiode werk je in project 56 aan een beroepsgerichte opdracht, waar je de kennis en vaardigheden uit deze module direct toepast. Dit project draait om professionele software-ontwikkeling. Bij deze ontwikkeling horen industriestandaard algoritmes. Daarnaast vraag het project om hoge performance bij algoritmes en zijn problemen vaak niet meer op de *naive* / ‘*brute* *force’* strategie op te lossen.

## Leermiddelen

Verplicht:

* Sedgewick, R., & Wayne, K. (2011). *Algorithms* (4th ed.). Addison-Wesley Professional
* http://docs.oracle.com/javase/tutorial/collections/

Facultatief:

* Introduction to Algorithms, T. H. Cormen, C. Stein, R. L. Rivest, C. E. Leiserson, The MIT Press, ISBN: 978-0-262-53305-8, 3de editie, 2009
* <http://stackoverflow.com/>
* http://docs.oracle.com/javase/tutorial/collections/index.html
* http://docs.oracle.com/javase/tutorial/java/generics/index.html
* Java Development Kit (JDK) versie 8, te downloaden van http://www.javasoft.com
* http://www.angelikalanger.com/GenericsFAQ/JavaGenericsFAQ.html

# Programma

Deze module kent een gecombineerde werkvorm waarin theorie en practicum afgewisseld worden.

In het theorie-deel wordt de theorie uit de literatuur kort samengevat. Je krijgt kleine oefenopdrachten of quizen om de leerstof toe te passen. Voor elke les bereid je je voor door de opgegeven leerstof te bestuderen. Tijdens de les kun je gericht vragen stellen.

In het practicum-deel pas je het geleerde toe door practicumopdrachten te maken. Je combineert hierin verschillende stukken theorie. De opdrachten maak je zelfstandig – tijdens het practicum en daarbuiten. In de les heb je de gelegenheid om je feedback te vragen aan de docent.

## 2.1 Onderwerpen

|  |  |  |
| --- | --- | --- |
| **Onderwerp** | **Inhoud** | **Literatuur** |
| Computationele complexiteitstheorie | * Computationele complexiteitstheorie   + Grote-O-notatie   + Empirische performance meting   + Memory | H1.4 |
| Fundamentele Datastructuren | * Abstract data types * Bags * Queues * Stacks | H1.3 |
| Sorteren | * Merge sort * Quick sort | H2 |
| Searching |  | H3 |
| Graphs | * Undirected * Directed * Shortest Path | H4 |

**Toetsing en beoordeling**

## 3.1 Procedure

De module wordt afgesloten met een practicumopdracht en een schriftelijk tentamen.

### 3.1.1 Normering

### 3.1.2 Tentamen

Het tentamen duurt twee lesuren. Bij het tentamen mag je 1 A4 aan zelfgemaakte, handgeschreven aantekeningen meenemen, dit is het enige toegestane hulpmiddel.

### 3.1.3 Practicum

Het programmeren voor de practicumopdracht wordt individueel of in duo’s uitgevoerd. Hierbij lever je de code in en een verslag, waarin je je keuzes met betrekking tot gebruikte datastructuren en algoritmes verantwoord. Het verslag maak je individueel en de deadline is op de dag van het tentamen.

## 3.2 Herkansing

Als een onderdeel (practicumopdracht en/of tentamen) een onvoldoende is, dan kun je dat onderdeel herkansen. In week 10 van de opvolgende onderwijsperiode in het hertentamen. De deadline voor de practicumopdrachtherkansing is in week 9 van de opvolgende onderwijsperiode.

## 3.3 Inzage

Het tentemen kun je inzien conform de regeling in de hogeschoogids.

**Bijlage 1. Toetsmatrijs**

|  |  |  |
| --- | --- | --- |
|  | **Leerdoelen** | **Vraag** |
| 1 | Je bent bekend met concepten van datastructuren en algoritmen | Tentamen |
| 2 | Je kan fundamentele datastructuren en algoritmes zelf implementeren | Practicumopdracht |
| 3 | Je kan de efficiency van algoritmen (tijd en resource) analyseren | Tentamen, Practicumopdracht |
| 4 | Je kan industriestandaard problemen behorende bij algoritmen herkennen en daar het juiste algoritme voor implementeren. | Practicumopdracht |