

Informaticawetenschappen III-InCo-d

Informatica- en communicatiewetenschappen

2023-05-30



Doel vak



Inleiding tot concepten van Informaticawetenschappen als voorbereiding op hoger onderwijs

Vanuit context: STEM

Met accent op:

- Algoritmen en programmeren
- Softwareontwikkeling
- Datacommunicatie, cloud en netwerksystemen

INZICHT



Aandachtspunten



Aandachtspunten

- Voorkennis leerlingen
- Leerplandoelen combineren bij het uitwerken van een concreet probleem
- Opbouwen van leerstof
- Vertrekken vanuit concrete STEM-problemen
 - STEM breed invullen: wetenschappen, technologie, engineering en wiskunde
- Linken aan leerplandoelen elektronica bv. input via sensoren op microcontroller
- Actuele leerinhouden



Algoritmiek



LPD 1 De leerlingen passen een gestructureerde programmeertaal toe bij het ontwerpen van een oplossing.

- Variabele, constante, gegevenstypes, operatoren, functies
- Gestructureerde programmeertaal - controlestructuren:
 - Sequentie
 - Selectie
 - Iteratie
- Gebruik van softwarebibliotheken:
 - Ingebouwde
 - Geïmporteerde



LPD 2 De leerlingen ontwerpen en implementeren algoritmen met datastructuren voor concrete problemen.

- Datastructuren:
 - ééndimensionale- en meerdimensionale arrays
 - lijst (list)
 - stapel (stack) - LIFO
 - wachtrij (queue) - FIFO
 - hashtable (hashtable)
- De meeste datastructuren zijn varianten van arrays. De manier waarop de array wordt benaderd of geïmplementeerd, bepaalt welke datastructuur het juist is. Hierbij wordt de datastructuur ofwel aangesproken via een index of via een sleutel.
- Je kiest de datastructuur in functie van de probleemstelling.



LPD 3 De leerlingen passen algoritmische technieken toe bij het programmeren van zelf ontworpen oplossingen.

- Algoritmische technieken
 - brute-force (Engels voor “brute kracht”)
 - dynamisch programmeren
 - gretig (greedy)
 - recursie
 - verdeel-en-heers (divide and conquer)
- Je kiest de algoritmische techniek in functie van de probleemstelling.



LPD 4 De leerlingen ontwerpen oplossingen die externe gegevens in- en uitvoeren.

- Externe gegevens = bestanden
- Verschillende types van bestanden:
 - CSV
 - TXT
 - JSON
 - XML
 - XLSX ...
- Toepassingen waarin je gaat sorteren, zoeken ... m.a.w. waarvoor je een grote hoeveelheid gegevens nodig hebt.
- Concrete oefeningen vanuit wiskunde, wetenschappen.



LPD 5 De leerlingen lossen concrete wiskundige problemen op met behulp van numerieke methodes

- Wat is een numerieke methode
- Hoe werkt de numerieke methode van een bepaalde wiskundige techniek
- Welke fouten kunnen optreden bij het gebruik van numerieke methodes
- Zelf programmeren of functie uit bibliotheek
- Koppelen aan wiskunde:
 - nulwaarden, extrema, afgeleiden, bepaalde integralen van functies,
 - matrixvermenigvuldiging, oplossen van stelsels eerstegraadsvergelijkingen.
- Bij voorkeur komen die numerieke methodes aan bod juist nadat die leerstof binnen de wiskunde is behandeld.



Voorbeeld 1

- Lees een getal in. Zoek of dit getal voorkomt in een gesorteerde rij.

Brute force

- Loop de rij af met een lus en vergelijk elk getal in de rij met het ingelezen getal tot je ofwel het getal gevonden hebt of het einde van de rij behaald hebt.
- Processortijd: lang als de rij veel getallen bevat en het ingelezen getal op het einde voorkomt

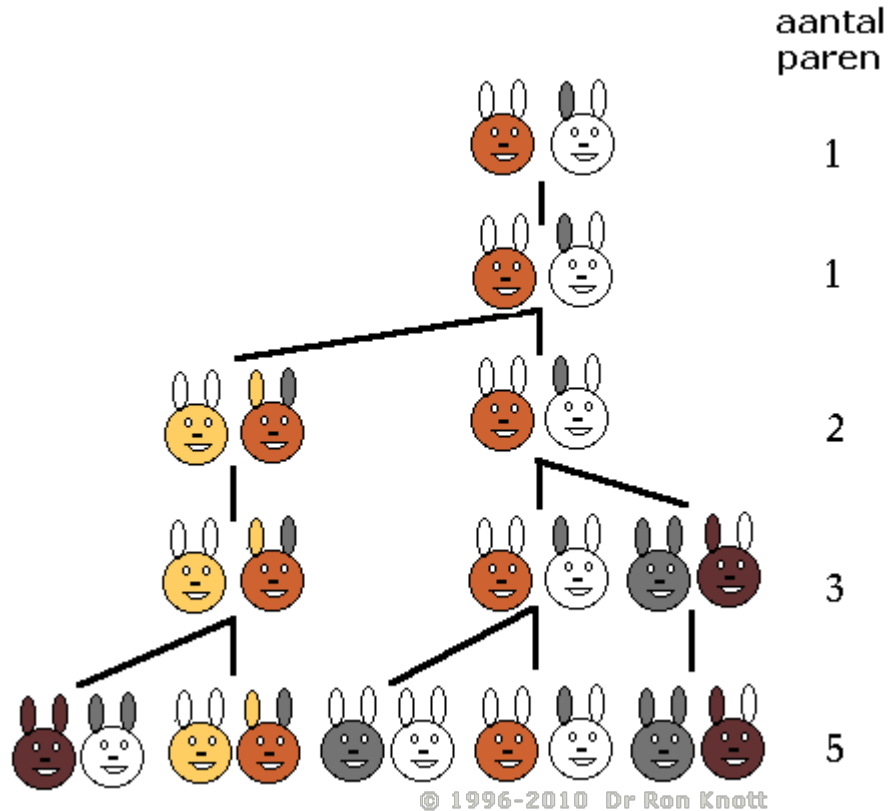
Verdeel en heers

- Zoek het midden van de rij. Vergelijk of het ingelezen getal kleiner, groter of gelijk is aan dit getal. Is het kleiner beperk de rij tot de eerste helft, is het groter beperk de rij tot de tweede helft. Herhaal tot het getal gevonden is of er geen rij meer is.
- Processortijd: korter, als de rij veel getallen bevat en het ingelezen getal op het einde voorkomt
Efficiëntere manier van werken



Voorbeeld 2

- Schrijf een programma dat het n-de Fibonacci-getal berekent, waarbij n wordt ingegeven.





Oplossing 1 - dynamisch programmeren

```
# bepaal het fibonaccigetel op een bepaalde positie

# de functie Fibonacci die het n-de Fibonacci-getal bepaalt
def Fibonacci(n):
    if n == 1 or n == 2:
        return(1)
    else:
        fibo1 = 1
        fibo2 = 1
        # bereken de volgende getallen uit de rij
        for teller in range(3, n + 1):
            fibo = fibo1 + fibo2
            fibo1 = fibo2
            fibo2 = fibo
        return (fibo)

# het programma

# lees de positie in
positie = int(input("Positie = "))

# print het fibonaccigetel uit
print(str(positie) + "e Fibonaccigetel =", Fibonacci(positie))
```

Positie = 15
15e Fibonaccigetel = 610



Oplossing 2 - recursie

Positie = 8
8e Fibonaccigetal = 21

$$F(5) = F(4) + F(3)$$

```
# bepaal het fibonaccigetal op een bepaalde positie met recursie

# de functie Fibonacci
def Fibonacci(getal):
    if getal == 1 or getal == 2:
        return 1
    else:
        return (Fibonacci(getal - 1) + Fibonacci (getal - 2))

# lees de positie in
positie = int(input("Positie = "))

# print het fibonaccigetal uit
print(str(positie) + "e Fibonaccigetal =", Fibonacci(positie))
```

Vergelijking algoritmes

Dynamisch programmeren

- Langer
- Geheugen: $1 + 3$
- Processortijd: de lus wordt n keer doorlopen.

```
Fibonacci_herh.py  
0.15625  
Positie = 40  
40e Fibonaccigetal = 102334155  
0.15625  
Dit programma duurt 0.0 seconden.
```



Recursie

- Korter
- Geheugen: $1 + n$ (aantal keer dat de functie wordt opgeroepen)
- Processortijd: elke functie roept 2 andere functies op, exponentieel

```
Fibonacci_recursie.py  
0.09375  
Positie = 40  
40e Fibonaccigetal = 102334155  
10.125  
Dit programma duurt 10.03125 seconden.
```



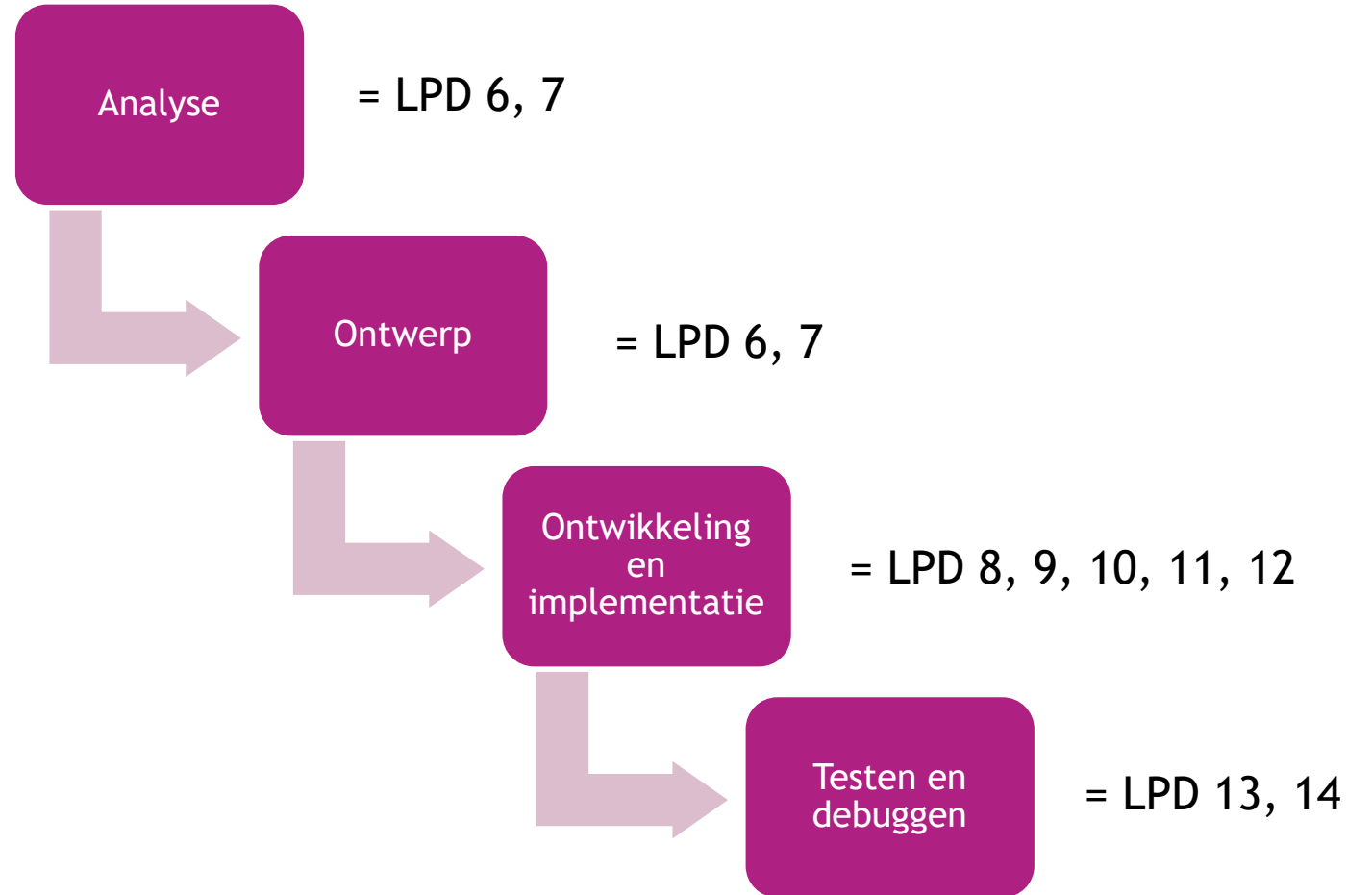


Efficiënt software ontwikkelen

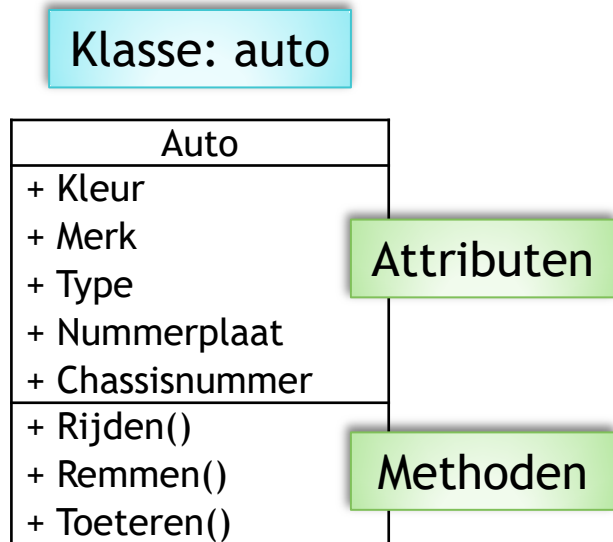


Softwareontwikkeling

Concreet probleem



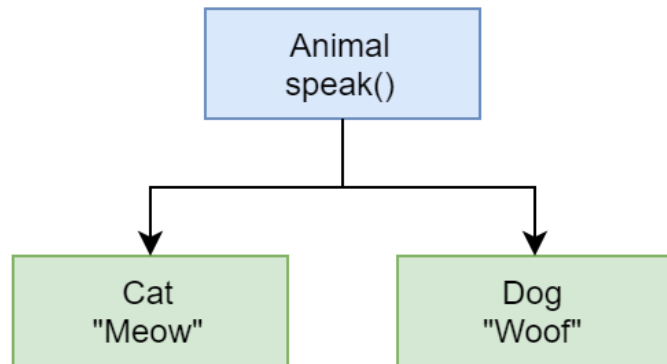
LPD 6 De leerlingen maken een ontwerp voor een softwaremodule en passen daarbij de regels van objectgeoriënteerd programmeren toe.



LPD 6 De leerlingen maken een ontwerp voor een softwaremodule en passen daarbij de regels van objectgeoriënteerd programmeren toe.

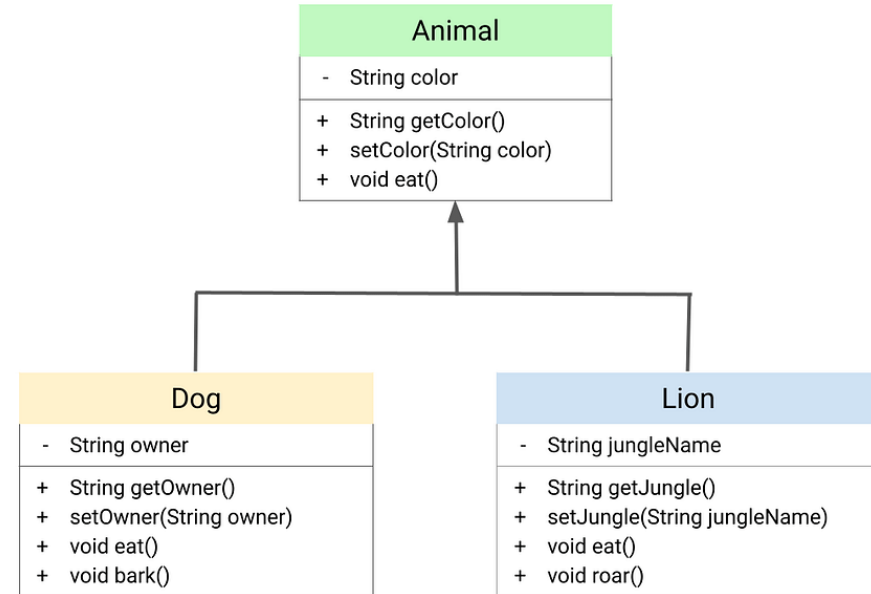
Principes van objectoriëntatie

Polymorfisme



Inkapseling

Overerving



Abstractie



LPD 7 De leerlingen maken een modulair ontwerp met een 3-lagenarchitectuur voor een softwaretoepassing en passen daarbij de basisprincipes voor een goed ontwerp toe.





Efficiënt software ontwikkelen

- LPD 8 De leerlingen implementeren objectgeoriënteerde softwaremodules op een manier die herbruikbaarheid en samenwerking vereenvoudigt.
 - Instantie, constructor
 - Commentaar
- LPD 9 De leerlingen houden bij het implementeren van softwaremodules rekening met gebruiksvriendelijkheid voor de eindgebruiker.
 - validatie, foutafhandeling, exception handling
- LPD 10 De leerlingen ontwikkelen een webinterface met hedendaagse webtechnologie.
- LPD 11 De leerlingen ontwikkelen client-side- of server-side scripts voor een modulair softwareontwerp.



Efficiënt software ontwikkelen

- LPD 12 De leerlingen hanteren aanwezige hulpbronnen bij het implementeren van softwaremodules en schrijven documentatie bij eigen ontworpen softwaremodules.
- LPD 13 De leerlingen testen eigen implementaties om eventuele fouten te identificeren en verwijderen.
 - Soorten fouten
- LPD 14 De leerlingen debuggen eigen implementaties om de bronnen van eventuele fouten op te sporen.



Datacommunicatie, computer- en netwerkachitectuur



Datacommunicatie, computer- en netwerkachitectuur

Computersystemen

- LPD 15 De leerlingen lichten de opbouw, de werking en de samenwerking toe van de componenten van een computersysteem.
 - Processor, intern geheugen, gegevens
 - Invoer, uitvoer, opslag
- LPD 16 De leerlingen maken een configuratie van een computersysteem op basis van gestelde eisen.

THEORETISCH



Datacommunicatie, computer- en netwerkkarchitectuur

Datacommunicatie en netwerksystemen

- LPD 17 De leerlingen lichten de opbouw en de werking toe van datacommunicatie en van een netwerk met zijn componenten en transportmedia.
 - OSI-referentiemodel
 - Communicatie
 - Adressering
- LPD 18 De leerlingen ontwerpen een configuratie van een netwerkstelsel op basis van gestelde eisen.

THEORETISCH

Netwerk = thuisnetwerk, een klein bedrijfsnetwerk, een kmo-netwerk



Datacommunicatie, computer- en netwerkkarchitectuur

Datacommunicatie en netwerksystemen

- LPD 19 De leerlingen beheren een computernetwerk.
 - Toegangs- en gebruikersrechten, gebruikersprofielen
 - Bronnen
 - DNS, DHCP
 - Shells scripting
- LPD 20 De leerlingen onderzoeken de oorzaken van dataverlies bij transport tussen netwerken en zoeken oplossingen aan.
 - Foutdetectie, foutcorrectie
- LPD 21 De leerlingen lichten het belang van encryptie toe en passen het toe in een netwerk.
- LPD 22 De leerlingen beveiligen een computernetwerk.

PRAKTISCH



Lesmateriaal en ondersteuning



Lesmateriaal

- Lesmateriaal:
 - Lesmateriaal screenen aan de hand van leerplandoelen
 - Datacommunicatie, computer- en netwerkachitectuur
 - [Sleutelboeken computerhardware - computernetwerken](#)
 - [Netwerken DEEL 1, 2 en 3 \(BoomBeroepsonderwijs Nederland\)](#)
 - [Netwerkbeheer met Windows Server 2019 deel 1 en 2 \(BoomBeroepsonderwijs Nederland\)](#)
 - [Leer werken met PowerShell \(versie 2021\) \(BoomBeroepsonderwijs Nederland\)](#)
 - Algoritmen en programmeren/Softwareontwikkeling
 - [Zie scherp scherper](#)
 - [Leren programmeren \(Acco - Dominiek Vandewalle\)](#)
 - [De programmeursleerling](#)
 - [Leren programmeren in C# Complete bundel, versie 2021 of verschillende delen \(BoomBeroepsonderwijs Nederland\)](#)
 - [Backend programmeren versie 2021 \(BoomBeroepsonderwijs Nederland\)](#)
 - Websiteontwikkeling
 - [Basis Webdevelopment \(BoomBeroepsonderwijs Nederland\)](#)



Lesmateriaal

- Naslagwerken:
 - [Handboek PowerShell vlot gebruiken](#) (Vanduuren Media - Frederik Vanhoo)
 - Algoritmen en programmeren - (Acco - Veerle Fack)
 - Webshop Katholiek Onderwijs Vlaanderen - (Buitengewoon) secundair onderwijs - leerboeken - Informatiebeheer



Lesmateriaal en ondersteuning

- Nascholing:
 - [Kickstart voor Informatica- en communicatiewetenschappen op 23/6/2023 in Howest in Kortrijk](#)
 - [Het nieuwe vak informaticawetenschappen: objectgeoriënteerd programmeren met C# op 22/6/2023 in Antwerpen](#)
 - Voor meer info over nascholingen: zie nieuwsbrief Informaticawetenschappen

Inschrijven nieuwsbrief:





Ondersteuning



Vragen en begeleiding

Greet Vanderbiesen

greet.vanderbiesen@katholiekonderwijs.vlaanderen

SAMENVATTING

