

# Activerende didactiek

*Inspiratiedag 25-03-2026*



***Learning by doing***  
informaticaActief

## Stichting informaticaActief

Online lesmateriaal voor informatica op havo en vwo.

- Ruim 40 cursussen
- Verschillende leerroutes
- Non-profit stichting
- Eigen cursus per lesgroep

INLOGGEN  
DIGITALE  
LESMETHODE



Voorlichting informatica



Onze ambitie



Agenda en planning



## Vacature bestuurslid

Wil jij bijdragen aan de toekomst van het informaticaonderwijs in Nederland? Stichting Informatica-Actief is op zoek naar een nieuw

Bekijk de  
vacature



Renske Weeda



Adriaan Dekker



Pieter Vorstenbosch



Paul Bergervoet

# Python Games

Doel van het **lesmateriaal**:  
leer Python programmeren en  
games ontwikkelen door te  
doen.

Doel **workshop**: lesmateriaal  
ervaren



Marten Postma

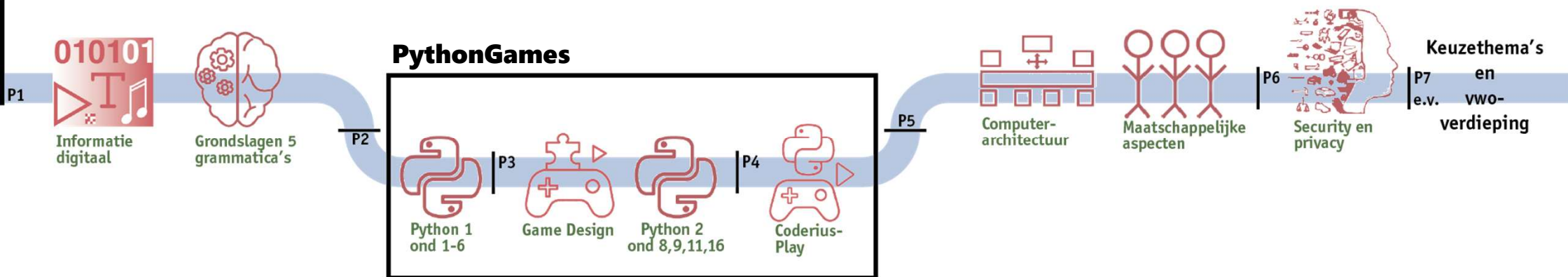


Renske Weeda



Koen Auke van Urk

# PyRoute



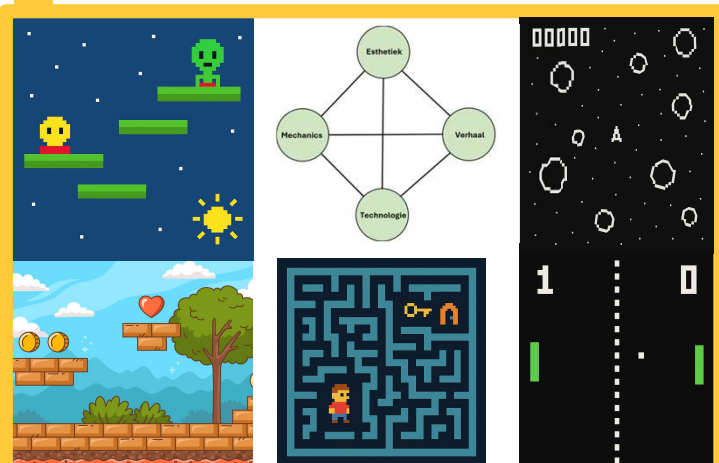


# PythonGames

Examenprogramma  
Domeinen D, P, C, B, F1

Verwerkingstijd  
Python 40 uren  
Play & Design 20 uren  
PO Eigen Game 20 uren

- > **Python:** variabelen, keuzen, functies, for-loops, lijsten, (geneste) dictionary's.
- > **Game Design:** spelervaring (esthetics, verhaal, mechanics, technologie), toestanden, usability, user experience.
- > **Coderius-Play:** vormen, mechanics.



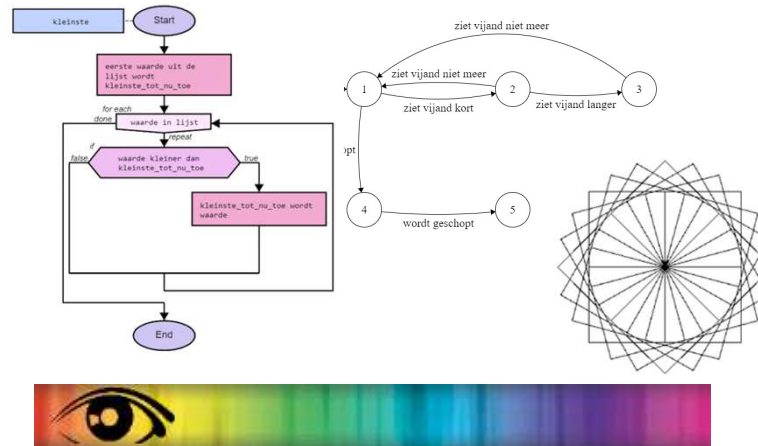
## Maak je eigen game in één geïntegreerde Python-track!

Theorie en praktijk lopen direct samen: eerst inoefenen met korte opdrachten, daarna toepassen in een eigen project. Zo komt kennis tot leven, inclusief thema's als grondslagen, informatie en interactie.

### Bijna alle kern subdomeinen:

- Code lezen, traceren en debuggen
- Ontwerp (automaten → programma)
- Standaardalgoritmen herkennen en gebruiken
- Codekwaliteit
- Informatie structureren en opslaan

Keuze domein: user experience



### Afsluitende PO

Ontwerp en implementeer je eigen game! Alle ontwerpaspecten komen aan bod: spelervaring, usability en toestandsdiagrammen. In de implementatie gebruik je alle geleerde Python-onderdelen: functies, lijsten, loops en dictionaries.

### Relatie met het examenprogramma

#### B - Grondslagen

- B1: Algoritmen (standaardalgoritmen m.b.t. geneste dictionaries)
- B2: Datastructuren (lijsten, (geneste) dictionary's)
- B3: Automaten (ontwerp van een game mbv. toestandsdiagram)

#### C - Informatie

- C1: Doelstellingen ((geneste) dictionary's en JSON)
- C2: Identificeren (gegevens identificeren voor game)
- C3: Representeren (lijsten, (geneste) dictionary's, JSON)
- C5: Gestructureerde data (geneste dictionary's)

#### D - Programmeren

- D1: Ontwikkelen (eigen game in Python maken)
- D2: Inspecteren en aanpassen (code lezen, aanpassen, uitbreiden – zowel in Python lesmateriaal als Play tutorials)

#### F - Interactie

F1: Usability (bestaande game analyseren a.d.h.v. heuristieken en toegankelijkheid; eigen game ontwerpen en analyseren)

#### P - User experience (keuzedomein)

- P1: Analyse (bestaande game analyseren adhv 4 pijlers)
- P2: Ontwerp (eigen game ontwerpen adhv 4 pijlers)

### Didactisch model

De stof wordt aangeboden op het moment dat deze relevant is, steeds afgewisseld met actief toepassen. Leerlingen ontwerpen en ontwikkelen hun eigen game en passen de theorie direct toe. Doordat ze hun eigen context kiezen en snel resultaat zien, blijft de motivatie hoog.

**Learning by doing** komt hier uitermate goed tot zijn recht.



### Ons didactische model:

- Learning by doing - praktisch en activerend
- Leren is stapsgewijs opgebouwd.
- Kennis is direct nodig voor de opdrachten, en uiteindelijk in de PO.



## PythonGames

### Examenprogramma

D – Programmeren

P – User Experience

C1 C2 C3 C5 – Informatie

B1 B2 B3 – Grondslagen

F1 - Interactie

### Verwerkingstijd

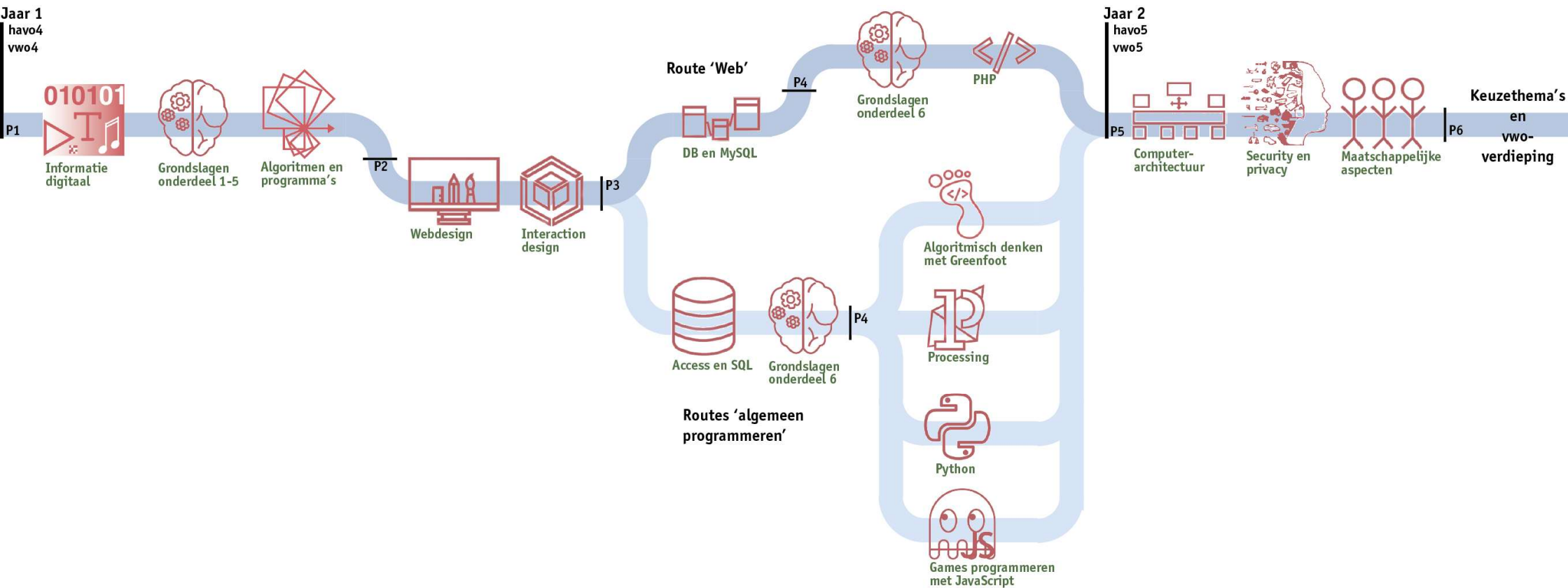
Python 40 uren

Play & Design 20 uren

PO Eigen Game 20 uren

- PRIMM-model
- Lesstof aangeboden wanneer relevant
- Debugging strategieën & Cheatsheets
- Activerende opdrachten met tips en antwoorden
- Deeltoets Python + Game opdracht incl. rubrics

# Tour vernieuwingen



# Security & Privacy



Examenprogramma

Domein A4, E2, F3 en F4

Verwerkingstijd

Basis 10 uur + 3 uur PO  
Verdieping 3 uur

- > **Online profiel**
- > **Betrouwbare communicatie**
- > **Ethiek en wetgeving**
- > **Bedreigingen**
- > **Authenticatie (verdieping)**



Deze cursus start met het verdienmodel achter gratis diensten door middel van tracking.

Daarna worden technieken behandeld die de betrouwbaarheid van een netwerk garanderen.

Voordat we ingaan op hacken, behandelen we eerst de ethische en juridische kaders.

Daarna worden social engineering, hack-technieken en allerlei vormen van malware uitgelegd.

De verdieping gaat in op authenticatie, omdat dat een belangrijke schakel is in de veiligheid van een systeem.



## Toetsing

Er is een theorietoets beschikbaar. Ook is er een opdracht waarbij leerlingen een informatieve poster of flyer maken over wat ze in deze cursus geleerd hebben.

## Relatie met het examenprogramma

### Domein A - Vaardigheden

- A4: Ethisch handelen

### Domein E - Architectuur

- E2: Security

### Domein F - Interactie

- F3: Privacy
- F4: Security

## Didactisch model

- Hands-on opdrachten om gevoel te krijgen bij bijvoorbeeld cookies en checksums.
- Opdrachten bij video's over social engineering en spyware.
- Casussen om in groepjes of klassikaal te behandelen.
- Nieuwsfeed van security.nl voor real-time actualiteit.

## Afronding

- Theorietoets
- PO poster of flyer

Wist je dat er ook een Priv/Sec compact cursus is?



### Ons didactische model:

- Learning by doing - praktisch en activerend
- Leren is stapsgewijs opgebouwd.
- Kennis is direct nodig voor de opdrachten, en uiteindelijk in de PO.



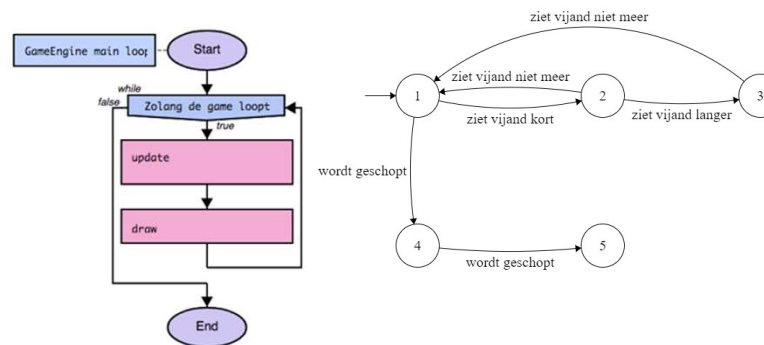
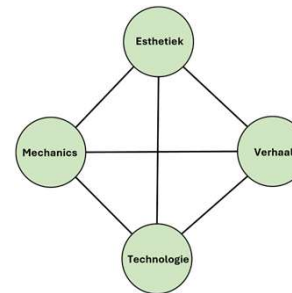
# Game Design

Examenprogramma  
Domeinen: P1, P2, F1, B3

Verwerkingstijd  
Basis 2 uren

## Inhoud

- > 1 : Inleiding Game Design
- > 2 : Spelervaring
- > 3: Esthetiek
- > 4: Verhaal
- > 5: Technologie
- > 6: Mechanics
- > 7: Mechanics - Toestanden
- > 8: Usability
- > 9 : Heuristieken voor usability
- > 10: Toegankelijkheid
- > 11: Eindopdracht



## Relatie met het examenprogramma Domein P - User Experience. Subdomein P1: Analyse

Analyse van een bestaande game a.d.h.v. de 4 pijlers van gaming theorie.

## Domein F - Interactie. Subdomein F1: Usability

Analyse van een bestaande gam a.d.h.v. heuristieken en toegankelijkheid.

## Domein B - Grondslagen. Subdomein B3: Automaten

Game ontwerp beschrijven m.b.v. een toestandsdiagram

## Didactisch model

Door zelf een bestaande (favoriete) game te analyseren leert de leerling nadenken over aspecten van spelervaring: esthetiek, verhaal, technologie en mechanics. Ook het ontwerp met toestandsdiagrammen en het analyseren van bruikbaarheid en toegankelijkheid komt aan bod. **Learning by doing** komt hier goed tot zijn recht.

## Afsluiting

Een presentatie met daarin de antwoorden op de opdrachten.

GAME DESIGN (OPDRACHT 7,3)

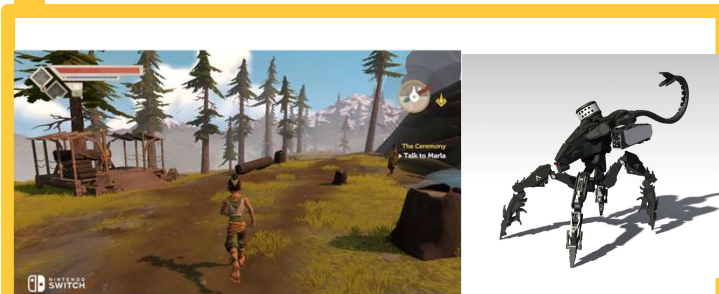
### TOESTANDSDIAGRAM

- Plak hier een screenshot/foto van je toestandsdiagram.
- Geef minstens 5 toestanden die het spel kan hebben.
- Op de pijlen aangeven waardoor de toestand verandert
- Tip: Tool voor toestandsdiagram: <https://madebyevan.com/fsm/>



## Eindopdracht

Gedurende de cursus maakt de leerlingen opdrachten waarbij ze een bestaande (zelf gekozen) game analyseren. Deze worden uitgewerkt in een presentatie.



## Ons didactische model:

- Learning by doing - praktisch en activerend
- Leren is stapsgewijs opgebouwd.
- Kennis is direct nodig voor de opdrachten, en uiteindelijk in de PO.



# SQL compact

Examenprogramma  
Domein: C1, C2, C3

Verwerkingstijd  
Basis 7 uren  
Verdieping 1 uren

> 1: Kennismaken met Databases

> 2: SELECT FROM

> 3: WHERE

> 4: LIMIT, ORDER BY en groepsfuncties

> 5: JOIN

> 6: Create - Read - Update - Delete

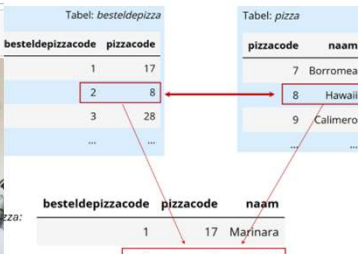
> 7: GROUP BY, HAVING

## Afsluitende Opdracht 4.13.1 De drie duurste pizza's

Toon van de drie duurste pizza's de naam en basisprijs zoals in het overzicht hiernaast.

naam	basisprijs
Combinazione	10.5
Specialità di Danilo	9.5
Calzone (dicke pizza)	9

```
SELECT *
FROM pizza
```



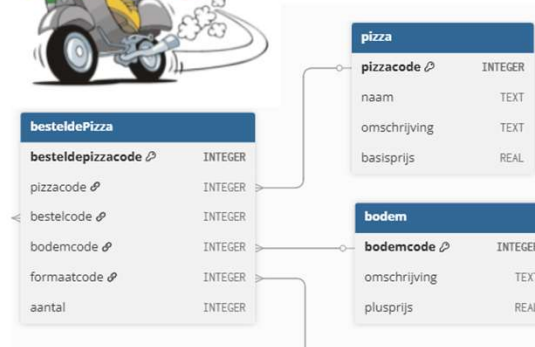
## Compacte SQL cursus

- **Theorie:** waarom databases onmisbaar zijn
- **Ontwerpen:** tabellen, ERD, datatypes en primary keys
- **Koppelen:** tabellen verbinden met foreign keys
- **Praktijk:** zelf SQL schrijven voor een SQLite-database

### Programmeeromgeving & context:

Direct in de browser — **geen** installatie of database-inrichting nodig.

De context is een herkenbare **pizzeria**, waarin ze bestellingen, klanten en menu's modelleren en gegevens opzoeken en presenteren.



Alle CRUD-aspecten (Create, Read, Update, Delete) komen aan bod. De opgedane kennis is direct inzetbaar, bijvoorbeeld voor een webshop in PHP of een Python-game met opslag van spelgegevens.

### Toetsing:

Afsluitend met een digitale of papieren toets, eventueel gecombineerd met een programmeeropdracht.

## Relatie met het examenprogramma

### Domein C: Informatie

#### Subdomein C1: Doelstellingen

De kandidaat kan doelstellingen voor informatie- en gegevensverwerking onderscheiden, waaronder zoeken en bewerken.

#### Subdomein C2: Identificeren

De kandidaat kan informatie en gegevens identificeren in contexten, daarbij rekening houdend met de doelstelling.

#### Subdomein C5: Gestructureerde data

De kandidaat kan een informatiebehoefte vertalen in een zoekopdracht op een verzameling gestructureerde data.

### Didactisch model

Leerlingen gaan vrijwel direct zelf aan de slag met het schrijven van SQL-code. Naast het opzoeken, aanpassen en verwijderen van data leren zij hoe ze tabellen kunnen ontwerpen, aanmaken, aanpassen en verwijderen. De opgedane kennis passen ze meteen toe in een eigen praktische opdracht (PO), bijvoorbeeld in combinatie met PHP of Python. **Learning by doing** komt hier goed tot zijn recht.



### Ons didactische model:

- Learning by doing - praktisch en activerend
- Leren is stapsgewijs opgebouwd.
- Kennis is direct nodig voor de opdrachten, en uiteindelijk in de PO.



# Architectuur compact

Examenprogramma  
Domein: E1

Verwerkingstijd

Theorie 4 uren

PO 2 uren

> 1 - Het drielagenmodel

> 2 - De fysieke laag

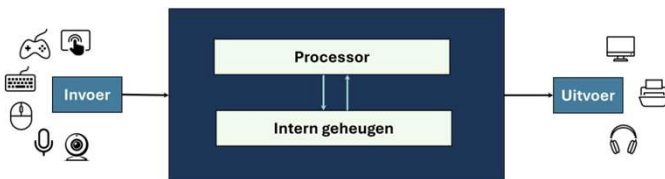
> 3 - De logische laag

> 4 - De toepassingslaag

> Afsluitende PO: Stel je eigen PC samen

Wat is het belangrijkste kenmerk van de Von Neumann-architectuur?

- De computer werkt online
- Er is geen invoer of uitvoer nodig
- Elke instructie heeft een eigen processor
- Gegevens en instructies staan in hetzelfde geheugen



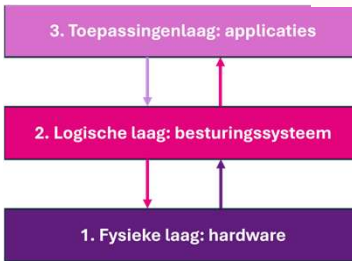
## Compacte cursus Architectuur

- **Theorie:** De focus ligt op het begrijpen en toepassen van het drielagenmodel.
- **Ontwerpen:**
  - Von Neumann-architectuur
  - drielagenmodel: fysieke laag, logische laag, toepassingslaag.
  - Samenwerking, diensten en communicatie van en tussen lagen
- **Praktijk:** zelf een PC samenstellen



### FYSIEKE LAAG

Componenten en hun functies



### Toetsing:

Tussendoor met formatieve toetsing: korte opdrachten en quizen.  
Afsluitend met een PO.

## Relatie met het examenprogramma

### Domein E: Architectuur

#### Subdomein E1: Decompositie.

De kandidaat kan de structuur en werking van digitale artefacten uitleggen aan de hand van architectuurelementen, dat wil zeggen in termen van de niveaulagen fysiek, logisch en toepassingen, en in termen van de componenten in deze lagen en hun onderlinge interactie.

#### Didactisch model

Leerlingen ontdekken stap voor stap hoe de die lagen, fysieke laag, logische laag en toepassingslaag, samenwerken om ervoor zorgen dat de computer, telefoon of ander digitaal apparaat goed werkt.

De opgedane kennis passen ze meteen toe in een eigen praktische opdracht (PO): 'Stel je eigen pc samen'. Leerlingen selecteren hardware en besturingssystemen voor een specifiek doeleinde binnen een budget. De beoordeling gebeurt via een uitgebreide rubric waarin onderbouwing, prijsbewustzijn en reflectie centraal staan. **Learning by doing** komt hier goed tot zijn recht.



#### Ons didactische model:

- Learning by doing - praktisch en activerend
- Leren is stapsgewijs opgebouwd.
- Kennis is direct nodig voor de opdrachten, en uiteindelijk in de PO.

# Codespaces

- Online VSC/Filius:
  - Github versiebeheer en samenwerken
  - Liveshare – live/gelijktijdig samenwerken
- Python: Turtle/Tkinter/SQLite/Play/PyGame-ce
  - <https://github.com/informatica-actief/python-template>
- Webontwikkeling: HTML/CSS/JavaScript/statische webserver
  - <https://github.com/informatica-actief/html-template>
- Netwerken: Filius
  - <https://github.com/informatica-actief/filius>

# Vacature: Bestuurslid bij *informaticaActief*

Wil jij bijdragen aan de toekomst van het informaticaonderwijs in Nederland?

Stichting *informaticaActief* is op zoek naar een nieuw bestuurslid dat wil meedenken en meewerken aan de verdere ontwikkeling van onze digitale leer methode voor het vak informatica.

Interesse? [bestuur@informatica-actief.nl](mailto:bestuur@informatica-actief.nl)



**Learning by doing**  
*informaticaActief*

# Vragen/opmerkingen