***Kraak jij de wiskundige en biologische code?***

**Bouwsteen 2**:Hoe kan je programmeren in Python? Hoe kan je een probleem in een online omgeving tot leven laten komen?

|  |
| --- |
| **Deze fase in een notendop:**  Binnen dit onderdeel ligt de focus op het programmeren in Python. Om de overgang hiernaar te maken is er een opdracht voorzien rond het determineren van bladeren. In deze opdracht wordt het eerder geziene stappenplan gekoppeld aan programmeren. Vervolgens gaan de leerlingen daadwerkelijk aan de slag met de programmeertaal Python. Hierbij is het de bedoeling dat klassikaal gestart wordt, maar de leerlingen doorheen de lesuren steeds meer zelfstandigheid krijgen. Dit doormiddel van het BBBB-systeem (Brain, Book, Buddy en Boss). Ter afsluiting is er een quiz waarbij de leerlingen in groepjes zowel hun kennis van het stappenplan als die van het programmeren moeten toepassen. |
| **Tijd**: 10 à 15 uur |
| **Leerdoelen**: De leerlingen kunnen:   * het stappenplan der computationeel denken toepassen op problemen; * basisfuncties programmeren in Python; * ALS/DAN-functies programmeren in Python; * booleaanse operatoren gebruiken in Python; * lussen programmeren in Python.   **STEM-doelen:**   * **LPD S2:** analyseren van natuurlijke en technische systemen aan de hand van verschillende STEM-concepten. * Je kan patronen laten ontdekken. * **LPD S8:** ontwikkelen van modellen om te visualiseren, te onderzoeken, op te lossen en te verklaren.   + Technologische modellen zoals schaalmodellen, technische tekeningen, algoritmes schema’s … |
| **Leerinhouden:** Python, CLI, editor, relationele operatoren, integer, float, string, print(), input(), variabelen, int(), float(), str(), type(), booleaanse expressie, if-statement, enkelvoudige keuzestructuur, if/else-statement, meervoudige keuzestructuur, statement, if/elif/else-statement, booleaanse operatoren, not, or, and, built-in-functies, while-lus, for-lus, in en range() |
| **Randvoorwaarden**:  **Materiaal voor klasgebruik:**   |  |  | | --- | --- | | Introcard |  | | Bladeren | Fotobehang Zeven verschillende bladeren op wit - PIXERS.BE | | Laptop | Surface Laptop Studio - Oneindig flexibel - Microsoft Surface | | Oefeningenblad |  | | Oplossingenbundel programmeren |  |   **Materiaal per groep voor quiz**:   |  |  | | --- | --- | | Quizformulieren |  |   **Voorkennis leerlingen**:   * De leerlingen kennen het stappenplan der computationeel denken en kunnen de verschillende stappen toepassen. * De leerlingen hebben al enige ervaring met het gebruik van een laptop.   **Externen:** Geen |
| Geen beschrijving beschikbaar.**Beschrijving leeractiviteiten**:  **Deel conceptenmap dat bij deze leeractiviteit hoort:**    **Overzicht leeractiviteit: timing + hoe te organiseren + hulpmiddelen**   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | **Beschrijving leeractiviteit** | **Duur** | **Hoe organiseren?** | **Hulpmiddelen** | | 1. | Introductie + inleidende oefening | 10’ | * Introcard   🡺 Bevraging vorige les   * Determineren van bladeren   🡺 Klassikale uitleg van de oefening  è In toegewezen groep aan de slag | - Introcard  - Bladeren  - Leerlingenbundel | | 2. | De basis in de programmeer-wereld | 20’ | * Klassikale inleiding van de programmeertaal en -omgeving * Bewerkingen worden zelf ontdekt * Relationele operatoren worden klassikaal aangebracht | - Leerlingenbundel  - Laptop | | 3. | Basisfuncties | 60’ | * Klassikale aanrijking met bijhorende oefenvoorbeelden   🡺 Ter ondersteuning:   * Bijhorende video’s * Individueel aan de slag met oefeningen (BBBB) | - Leerlingenbundel  - Laptop | | 4. | ALS/DAN-functies | 120’ | * Begeleid zelfstandig werk (BBBB)   🡺 Ter ondersteuning:   * Bijhorende video’s * Extra uitleg | - Leerlingenbundel  - Laptop | | 5. | Lussen | 120’ | * Klassikale aanrijking van de while-lus en keyword ‘in’   🡺 Ter ondersteuning:   * + Bijhorende video’s * Individueel aan de slag met de theorie van de for-lus en de functie range()   🡺 Ter ondersteuning:   * + Bijhorende video * Individueel aan de slag met de oefeningen (BBBB) | - Leerlingenbundel  - Laptop | | 6. | Overkoepelende oefeningen | 300’ | * Individueel aan de slag met oefeningen (BBBB)   🡺 Eigen tempo  è Eigen volgorde | - Leerlingenbundel  - Laptop | | 7 | Quiz | 45’ | * Inhoudelijke quiz over het stappenplan der computationeel denken en programmeren   🡺 In toegewezen groepen (3 à 4 lln)  è Meerkeuzevragen | - Quizformulieren  - PowerPoint  - Eventuele prijs voor winnende groep |   **Extra uitgebreide info bij leeractiviteit:**   1. **Introductie + inleidende oefening**   Als introductie wordt een introcard afgenomen. Hiermee toetst de leerkracht de kennis van voorgaande les(sen) af. Vervolgens volgt een inleidende oefening op het eigenlijke programmeren. Om de link met de biologie te bekrachtigen, gaan leerlingen aan de slag met echt materiaal uit de natuur. Het is namelijk de bedoeling om bladeren te determineren a.d.h.v. een determinatiekaart. In wat volgt wordt het verband tussen het handmatig determineren en determinatie-apps verduidelijkt. Dit zorgt voor een mooie overgang naar de digitale wereld.   1. **De basis in de programmeerwereld**   In dit onderdeel komen leerlingen een eerste keer in aanraking met de programmeeromgeving. Deze verkenning gebeurt klassikaal waarna de leerlingen op zelfstandige basis de bewerkingen (som, verschil, …) in Python ontdekken. Verder volgen nog enkele oefeningen omtrent de bewerkingen en relationele operatoren (<, >, …). Bij het oefenen van nieuwe leerstof wordt vaak de nadruk gelegd op het ‘debuggen’. Het zal belangrijk zijn om ze al zo vroeg mogelijk te wijzen op mogelijke fouten. Dit zal ervoor zorgen dat het opsporen van fouten in latere opdrachten vlotter verloopt. Echter kunnen we nooit volledig voorkomen dat er fouten gemaakt worden. Zo wordt er enige vorm van zelfstandigheid gevormd omtrent het ‘debuggen’. Wanneer leerlingen op eigen tempo oefeningen maken, wordt het BBBB-systeem nagestreefd.   1. **Basisfuncties**   De basisfuncties in Python worden klassikaal aangebracht. Bij iedere basisfunctie hoort een oefenvoorbeeld. De leerkracht laat dit oefenvoorbeeld, bij de klassikale aanbreng, door de leerlingen programmeren. Op deze manier hebben ze de basis al een beetje onder de knie en kunnen ze hierop beginnen verder bouwen. Bovendien is deze korte inoefening ook nuttig daar er al heel wat aan foutenanalyse gedaan kan worden. Eens de basisfuncties doorlopen zijn, mogen de leerlingen van start gaan met oefeningen. Deze maken ze individueel. Er wordt nog steeds rekening gehouden met het BBBB-systeem.   1. **ALS/DAN-functies**   De ALS/DAN-functies worden door de leerlingen zelfstandig doorlopen. De leerlingenbundel is steeds voorzien van voldoende ondersteuning. Zo kunnen ze gebruik maken van uitgeschreven theorie, video’s en oefenvoorbeelden. Doorheen de theorie staat bovendien ook een interessant weetje, dat de leerlingen kunnen lezen bij interesse.  Naast de ALS/DAN-functies worden de booleaanse operatoren ook behandeld in dit leerstofonderdeel. Leerlingen onderzoeken zelf de hiërarchie van de operatoren in een oefening. Ook deze oefeningen worden zelfstandig doorlopen (BBBB-systeem).   1. **Lussen**   Bij het deel van de lussen wordt eerst de while-lus en het keyword ‘in’ klassikaal besproken. Ter ondersteuning zijn er ook video’s beschikbaar. Deze kunnen de leerlingen als back-up gebruiken. Vervolgens nemen ze de for-lus zelf onder handen a.d.h.v. een bijhorende video. In wat volgt zullen ze aan de slag gaan met de oefeningen (BBBB-systeem).   1. **Overkoepelende oefeningen**   In dit onderdeel gaan de leerlingen aan de slag met de overkoepelende oefeningen. Deze behandelen alle voorgaande geziene theorie. Leerlingen mogen zelf kiezen welke oefeningen en in welke volgorde ze maken. Ze bepalen als het ware hun eigen leerproces. Ook in deze oefeningenreeks zijn er interessante biologische weetjes.   1. **Quiz**   Afsluiten doen we met een algemene quiz die in groepjes van 3 à 4 leerlingen gespeeld wordt. Aan de hand van 20 meerkeuzevragen wordt de kennis over zowel het stappenplan als het programmeren getest. Wanneer alle vragen overlopen zijn, controleren de groepjes elkaars antwoorden. Voor de groep met de hoogste score kan eventueel een prijs voorzien worden. Indien er twee groepen met eenzelfde score zijn, is er een schiftingsvraag voorzien namelijk: “Wie kan het meeste cijfers na de komma van het getal pi geven?”.  **Extra info: volgsysteem**  Om het klasoverzicht te bewaren en ervoor te zorgen dat de leerlingen niet te ver uiteen lopen, maar elkaar ook niet ophouden, werken we met een volgsysteem. Hierbij plaatsen we de titels van de programmeeronderdelen (zie foto hieronder) op het bord, met onder het eerste onderdeel de namen van alle leerlingen. Binnen elk onderdeel worden een aantal verplichte oefeningen geselecteerd. Wanneer een leerling deze oefeningen af heeft, verplaatst hij zijn naam op het bord naar het volgende onderdeel en begint hij aan de niet verplichte oefeningen. Als alle namen naar een volgend onderdeel verplaats zijn, beginnen de leerlingen aan de verplichte oefeningen van dit onderdeel. Leerlingen die eerder klaar zijn met alle oefeningen mogen al verder werken.  No description available. |
| **Ondersteunend materiaal voor leerlingen en leerkrachten**:  *Dit zijn verdere verwijzingen naar concreet lesmateriaal voor leerlingen, zoals werkblaadjes, en voor leerkrachten, zoals PowerPoint presentaties. Mogelijks zal hier al naar verwezen worden in de beschrijving leeractiviteiten. Bedoeling is dat je hier een link maakt naar de respectievelijke werkblaadjes, presentaties, etc.*  *Cursus voor de leerkracht: Kraak jij de wiskunde en biologische code? (handleiding + PowerPoint) + handboek ‘programmeren in Python’*  *Werkbundel: leerlingenbundel*  *Programmeeromgeving:* [*https://www.programiz.com/python-programming/online-compiler/*](https://www.programiz.com/python-programming/online-compiler/)  *Ondersteunende video’s:* <https://www.youtube.com/watch?v=7f90IFmsj6o&list=PLb7wduIUUvjWIRhtPIFC6S8lrm9kw8D-p> |
| **Reader**:  Dit zijn verwijzingen naar voor de leerkracht interessante bronnen over deze bouwsteen met extra achtergrondinformatie (filmpjes, boeken, artikels, websites, etc.)  Inleiding + algemene oefening   * <https://www.plantyn.com/webshop/product/hoezo-1-23u-leerwerkboek-(incl.-scoodle)-9789030152781> * <https://www.libelle.be/nest/apps-planten-herkennen/> * <https://www.wandel.nl/uitrusting/5-gratis-apps-om-dieren-en-of-planten-te-herkennen/> * <https://leafsnap.app/>   Basis in de programmeerwereld   * <https://www.programiz.com/python-programming/online-compiler/> * <https://www.youtube.com/watch?v=7f90IFmsj6o&list=PLb7wduIUUvjWIRhtPIFC6S8lrm9kw8D-p> <https://vimeo.com/544984963> * <https://www.youtube.com/c/UHasseltTutorials/videos> * <https://dodona.ugent.be/> * <https://www.pelckmans.be/programmeren-met-python-werkboek-incl-pelckmans-portaal.html>   Basisfuncties   * <https://www.programiz.com/python-programming/online-compiler/> * <https://www.youtube.com/watch?v=7f90IFmsj6o&list=PLb7wduIUUvjWIRhtPIFC6S8lrm9kw8D-p>   ALS/DAN-functies   * <https://www.programiz.com/python-programming/online-compiler/> * <https://www.youtube.com/watch?v=7f90IFmsj6o&list=PLb7wduIUUvjWIRhtPIFC6S8lrm9kw8D-p> * <https://nl.wikipedia.org/wiki/George_Boole> * <https://plato.stanford.edu/entries/boole/>   Lussen   * <https://www.programiz.com/python-programming/online-compiler/> * <https://www.youtube.com/watch?v=7f90IFmsj6o&list=PLb7wduIUUvjWIRhtPIFC6S8lrm9kw8D-p>   Overkoepelende oefeningen   * <https://www.pelckmans.be/programmeren-met-python-werkboek-incl-pelckmans-portaal.html> * <https://youtu.be/36kxhO_ki8o> * <https://dodona.ugent.be/nl/activities/933531418/> * <https://geocachen.be/geocaching/geocache-puzzels-oplossen/rij-van-fibonacci/> * <https://dodona.ugent.be/nl/activities/1593933361/>   Quiz   * <https://pynative.com/basic-python-quiz-for-beginners/?fbclid=IwAR2vEUf3uPPPYkrkVZ8VoF6bIu9SX4d_-zal7p3--bmohHbWmtton9ULSRc> * <https://www.w3schools.com/quiztest/quiztest.asp?qtest=PYTHON>   **ICT-tools:** YouTube |
| **Wiskunde** **Leerplannen:**  **LPD 2:** De leerlingen onderzoeken interacties tussen wiskunde en andere domeinen via wiskundige toepassingen.  **LDP 60:** De leerlingen ontwerpen algoritmes om problemen digitaal op te lossen. **Eindtermen:**  **ET. 4.5:** De leerlingen ontwerpen algoritmen op problemen digitaal op te lossen. Met inbegrip van kennis. **Biologie** **Leerplan eerste graad:**  **LPD 9:** De leerlingen onderzoeken via een terreinstudie voor een biotoop de onderlinge afhankelijkheid van verschillende organismen en de rol van biotische en abiotische factoren.   * Determinatie van planten  **Leerplan derde graad:**  **LPD B23:** De chemische structuur van het DNA en RNA bespreken.  **LPD B38:** Aan de hand van concrete vraagstukken, kruisingsschema's en stambomen, de overerving van codominante allelen, letale allelen, multipele allelen, polygenie, geslachtsgebonden allelen, gekoppelde genen en recombinatie van gekoppelde genen interpreteren en toelichten. **ICT**  **LPD 3:** De leerlingen gebruiken doelgericht en adequaat standaardfunctionaliteiten van digitale infrastructuur en online en offline toepassingen om digitale inhouden te beheren.  **GLI-ddaa 9:** De leerlingen gebruiken doelgericht en adequaat standaardfunctionaliteiten van digitale infrastructuur en toepassingen om digitale rekenbladen te creëren. **STEM**  **LPD S2:** De leerlingen analyseren natuurlijke en technische systemen aan de hand van verschillende STEM-concepten.  **LPD S8:** De leerlingen ontwikkelen modellen om te visualiseren, te onderzoeken, op te lossen en te verklaren. |
| Geen beschrijving beschikbaar.  **Ontwikkeld in samenwerking met:** Sint-Jozefinstituut Bokrijk |