|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Situering** | **Opleiding**:  ICT programmeren | | **Toepassingsgebied**:  (Kies hieronder het toepassingsgebied/de doelgroep voor wie deze taak bedoeld is)  Vrije tijd | |
| **Module**:  Start to program | **Vermoedelijke aantal lestijden van de ICT-taak**:  4 - 6 lestijd(en) | | **Auteur(s) & CVO**:  geert.linthoudt@pcvodenderenschelde.be |
| **Titel van de ICT-taak**:  Scoreverloop (van een tenniswedstrijd). | | | |
| **In te oefenen basiscompetenties van deze ICT-taak (schrap de BC's die niet in de authentieke taak zitten):**   * ~~IC BC013 - \* gaat bewust en kritisch om met digitale media en ICT~~ * ~~IC BC017 - kan ICT veilig en duurzaam gebruiken~~ * IC BC023 - kan ICT aanwenden om problemen op te lossen * ~~IC BC024 - \* kan zijn eigen deskundigheid inzake ICT opbouwen~~ * IC BC228 - kent de verschillende principes en onderdelen op basis waarvan een programma kan opgebouwd worden * IC BC229 - begrijpt de basisprincipes van programmeren * ~~IC BC230 - heeft inzicht in de processen achter het computationeel denken~~ * IC BC237 - kan een eenvoudig programma wijzigen * IC BC239 - kan bij het programmeren een juiste logica volgen * IC BC240 - kan een eenvoudig programma maken * IC BC242 - kan een programma uittesten * IC BC243 - kan een programma documenteren * ~~IC BC258 - houdt rekening met regelgeving m.b.t. licenties voor het gebruik en de publicatie van broncode~~ * ~~IC BC288 - kan ICT-problemen oplossen~~ * IC BC344 - kan oplossingen voor eenvoudige probleemstellingen bedenken * ~~IC BC349 - \*is nieuwsgierig naar de mogelijkheden van ICT op het vlak van programmeren en het creëren van output~~ * IC BC352 - begrijpt het systeem en de functies die achter een programma zitten | | | |
| **Omschrijving** | **Concrete case of probleemstelling**:  U speelt wel eens een wedstrijdje tennis en u wilt een klein programmaatje maken dat het scoreverloop van een webstrijd kan bijhouden.  **Inleiding**  De cursist heeft reeds kennis van basisbegrippen van het programmeren en heeft inzicht in selecties en iteraties. | | | |
| **Lesverloop/stappenplan** | **De effectieve ICT-taak**:  Ontwikkel een algoritme of een klein programmaatje dat het scoreverloop van een tenniswedstrijd bijhoudt. U hebt hiervoor 2 knoppen. Een knop voor speler 1 en een knop voor speler 2. Bij het klikken op de specifieke knop wordt de score van die speler verhoogd en weergegeven.  **Lesverloop/stappenplan**:   |  |  | | --- | --- | | **Opdrachten** | **BC** | | Teken of ontwerp de lay-out die minimaal moet bestaan uit 2 knoppen en 2 labels (om de stand in weer te geven). | IC BC228 | | Ontwerp het algoritme (stroomschema/pseudocode/programma) dat het scoreverloop bijhoudt. | IC BC023  IC BC229  IC BC239  IC BC240  IC BC243 | | Test het algoritme uitvoerig om te zien of het effectief klopt. Pas het algoritme waar nodig aan. | IC BC237  IC BC242 | | Optioneel: voeg een invoercorrectie toe (voor het geval u een punt aan de verkeerde speler hebt toegekend) | IC BC237 | | | | |
| **Bronnen** | **Bronnen**:  <https://ictopleidingen.azurewebsites.net/start-to-program-de-anatomie-van-een-computerprogramma/>  <https://ictopleidingen.azurewebsites.net/start-to-program-control-flow/>  <https://nl.wikipedia.org/wiki/Tennis> | | | |
| **Richtlijnen** | **Extra leerkracht informatie**  De nadruk bij deze authentieke taak ligt op het correct gebruik van selecties.  U kunt u beperken tot het uitwerken van het algoritme in een stroomdiagram of in pseudocode. Eventueel kunt u het later effectief programmeren in een specifieke ontwikkelomgeving (in de module “Specifieke ontwikkelomgeving – Eenvoudige functionaliteiten).  Dit algoritme kan eventueel nu reeds geprogrammeerd worden in een eenvoudige ontwikkelomgeving (bv. App Lab <https://code.org/educate/applab> ).  **Spelregels**  De punten worden geteld volgens een systeem 15, 30, 40, game, set en match.  Een *match* of wedstrijd wordt gespeeld naar twee gewonnen sets.  Een [*set*](https://nl.wikipedia.org/wiki/Set_(tennis)) wordt gewonnen door de speler die het eerst 6 games wint, mits het verschil minimaal twee games bedraagt. Als de stand in een set 6-5 is en de speler die op 6 staat de volgende game wint, dan wordt het 7-5 en is de set afgelopen. Wordt het echter 6-6 dan zijn er twee mogelijkheden waarbij de tiebreak de meest gebruikelijke is:   * doorspelen totdat er een verschil van twee games wordt bereikt. * er wordt een [*tiebreak*](https://nl.wikipedia.org/wiki/Tiebreak_(tennis)) gespeeld. Dit is te beschouwen als een bijzondere game en de winnaar van deze tiebreak wint de set.   Een [*game*](https://nl.wikipedia.org/wiki/Game_(tennis)) wint men door vier gewonnen punten, die geteld worden als 15, 30, 40 en 'game'. Nul punten duidt men in het Engels traditioneel aan met "love". Ook hier echter een verschil van twee. Als het 40-40 (deuce) wordt dan worden er nog minimaal 2 punten gespeeld. Eerst heb je voordeel (voor de serveerder) of nadeel (voor de ontvanger) en dan game, Het kan lang duren tot de game uiteindelijk gewonnen wordt, doordat het na voordeel of nadeel steeds weer deuce kan worden. | | | |