

Randvoorwaarden programmeerplatform “Computational Thinking” onderbouw VO

Benny Aalders

Vincent Velthuisen

15 juni 2017

Samenvatting

In de SLO leerdoelen voor Digitale Geletterdheid staan vaardigheden beschreven. Voor het verwerven van die vaardigheden wordt een platform gebruikt. Dit document beschrijft de randvoorwaarden waaraan dit platform, met name voor de vaardigheden in het domein Computational Thinking, moet voldoen.

Bij de voorwaarden is een mogelijke invulling aangegeven. Dit is echter geen bindend voorschrift voor hoe aan de voorwaarde voldaan dient te worden.

Inhoudsopgave

1	Voorwaarden	2
2	Huidige stand van zaken	3
2.1	Toelichting	3
2.1.1	Casio	3
2.1.2	HP Prime	4
2.1.3	TI	5

Begrippenlijst

platform verwijst in deze context naar het apparaat waarop de leerling zijn of haar code schrijft en uitvoert. Dit kan een (grafische) rekenmachine zijn, maar ook een telefoon, tablet, laptop of desktop computer.. 2, 3

testserie een lijst met mogelijke invoeren voor een programma en het verwachte antwoord. Kan gebruikt worden voor het (automatisch) beoordelen van software.. 2

Inleiding

SLO werkt momenteel aan de leerlijnen voor Digitale Geletterdheid in de onderbouw van het voortgezet onderwijs. Onderdeel van deze leerlijnen is het vermogen van de leerlingen om zelf code te schrijven. Het schrijven van de code zal plaatsvinden op een bepaald platform. Dit document beschrijft de voorwaarden waaraan een potentieel programmeerplatform moet voldoen om in de leerlijnen te passen.

Een potentieel platform is een grafische rekenmachine. Leerlingen zullen in de bovenbouw sowieso genooddaakt zijn een dergelijk apparaat aan te schaffen volgens de richtlijnen van school. Het kan voor scholen aantrekkelijk zijn om dit platform al in de onderbouw te gaan gebruiken. Dit document probeert overzichtelijk te maken in welke mate de grote producenten van grafische rekenmachines al voldoen aan de voorwaarden die in dit document gesteld worden aan een programmeerplatform.

1 Voorwaarden

Tabel 1: Overzicht van de randvoorwaarden en mogelijke invulling daarvan

#	Voorwaarde	Uitleg en mogelijke invulling
1	Het platform moet programmeerbaar zijn, bij voorkeur zonder tussenkomst van een ander apparaat.	Het moet voor leerlingen gemakkelijk zijn om bestaande code te bestuderen en er mee te experimenteren. Ook tijdens de les. Dit gaat moeilijker als daarvoor een aparte computerruimte nodig is.
2	Beschikbaar maken van voorbeeldcode en hulpcode aan leerlingen wanneer de docent dat wil.	Een softwarebibliotheek die beschikbaar is vanaf het platform van de leerling en waarvan de docent kan bepalen welke elementen wel of niet beschikbaar/zichtbaar zijn.
3	Inleveren van de geschreven opdrachten.	Een database waarin de leerling zijn of haar opdracht kan inleveren.
4	(Automatisch) controleren van en feedback geven op de code van de leerlingen.	De docent moet de ingeleverde code gemakkelijk kunnen testen met behulp van testseries. Het deels kunnen automatiseren van deze controle is een pre.
5	Het gedrag van leerlingen kunnen controleren.	Live kunnen meekijken op het platform van de leerling om voortgang te kunnen zien en eventueel veel voorkomende problemen te herkennen. Ook om te kunnen controleren of er geen fraudulent gedrag is tijdens examinering.

Tabel 1: Overzicht van de randvoorwaarden en mogelijke invulling daarvan

#	Voorwaarde	Uitleg en mogelijke invulling
6	De staat van het geheugen van het platform kunnen beïnvloeden en uitlezen.	Aan het begin van een toets mag slechts bepaalde informatie aanwezig zijn in het geheugen van de rekenmachine. Aan het einde van de toets moet de staat van het geheugen kunnen worden vastgelegd voor becijfering. Werk van de leerling dat aanwezig was voor het starten van de toets moet na de toets weer beschikbaar zijn voor de leerling, maar niet tijdens de toets.
7	Voordragen nieuwe code voor de bibliotheek.	Leerlingen kunnen eigen code voordragen voor de bibliotheek, bijvoorbeeld voor gebruik tijdens een toets. De code kan, naar keuze van docent, beschikbaar gemaakt worden voor de leerling, de klas of de jaargroep.

2 Huidige stand van zaken

Tabel 2 geeft aan in welke mate de grootste fabrikanten van grafische rekenmachines al voldoen aan de voorwaarden.

Tabel 2: Huidige stand van zaken

#	Casio		HP	TI	
	fx-CG20	fx-CG50	Prime	84+	Nspire
1	✓	✓	✓	✓	✓
2	extern	extern	✓	✓	✓
3	extern	extern	✓	✓	✓
4	?	?	✓	?	?
5	✗	?	✓	✗	✗
6	✓	✓	✓	✓	✓
7	extern	extern	✓	✓	✓

2.1 Toelichting

2.1.1 Casio

1. De fx-CG20 en fx-CG50 zijn programmeerbaar in een aangepaste BASIC versie.
2. Dit kan natuurlijk in een Drive. Dat kan een school het beste zelf opzetten.

3. Programma's zijn exporteerbaar. Programma's hebben extensie .g3m. Geëxporteerde bestanden kunnen bijv. in Word worden aangepast en weer worden geïmporteerd. De op te zetten database kan de school zelf opzetten.
4. Programma's kunnen tabellen aanroepen en gebruiken als invoer.
5. De Casio fx-CG20 heeft geen live verbinding met een master computer o.i.d. Tijdens examinering kan de examenstand niet gebruikt worden. Het CvTE eist dat dan programmeren niet mogelijk is. Wel kan de machine simpel worden gereset met de toetsencombinatie 1-3-ON. Dan zijn alle gegevens gewist.
6. Onze machine heeft flashgeheugen en programma's kunnen eenvoudig heen en weer gesleept worden in Windows. Ook kun je programma's op slot zetten met een password. Het is natuurlijk wel zo dat als je een machine leeg maakt voor een toets je een backup moet maken of de programma's elders op moet slaan wil je ze bewaren.
7. Natuurlijk kan dit, maar niet automatisch via een draadloze verbinding.

Natuurlijk is de gebruiksvriendelijkheid van de machine en de beschikbare programmeertaal ook belangrijk in dit verband. Op Youtube, waar ze de emulator gebruiken om goed te kunnen filmen, staan voorbeelden. <https://www.youtube.com/watch?v=4bBP5Bo4Eco>

Ook leuk is het programmeren van de GR zelf. Dus manipulaties op de OS en dergelijke. Hierover zijn veel forums en docu. Dit is een soort startpagina (PRIZM is de Amerikaanse naam voor de fx-CG20). http://prizm.cemetech.net/index.php/Prizm_Programming_Portal

Hier wat voorbeelden van wat er al gemaakt is. <https://www.cemetech.net/programs/index.php?mode=folder&path=/prizm/>

2.1.2 HP Prime

1. De HP Prime kan in HPPL (HP Programming language) worden geprogrammeerd, een Pascal-achtige taal.
2. Mogelijk op verschillende manieren: advies is om het (draadloze) klassennetwerk te gebruiken voor het wel of niet beschikbaar maken van elementen.
3. Programma's kunnen bijvoorbeeld als bijlage aan een email worden gekoppeld, de programmatekst kan in een Word document worden geplakt.
4. Als het programma de juiste naam krijgt, kan relatief eenvoudig een testprogramma worden geschreven dat deze test uitvoert.
5. Mogelijk met het klassennetwerk.
6. Eenvoudig mogelijk via de configureerbare examenstand.
7. Ieder programma kan andere programma's aanroepen, door een set programma's voor te laden in het geheugen van alle te gebruiken machines (via klassennetwerk) kan dit eenvoudig worden gerealiseerd.

2.1.3 TI

1. TI Basic is beschikbaar op de 84+ en de Nspire. De Nspire kan ook Lua scripts interpreteren.
2. Beide platformen hebben desktopsoftware waarmee gecommuniceerd kan worden. Voor de Nspire familie is ook een docking station beschikbaar.
3. Programma's kunnen middels de desktopsoftware op een pc bewaard worden.
4. Via bovengenoemde mechanisme kan de code gedeeld worden met de docent.
5. Niet van toepassing op TI.
6. Examenstand.
7. De 84 heeft hier geen ingebouwde technieken voor maar het kan wel via eerder genoemde desktop software. De Nspire heeft beschikking over Private en Public bibliotheken.

De TI grafische handhelds kunnen verbonden worden met de TI-Innovator™ Hub. Dit stelt leerlingen in staat om de basis van coderen en ontwerpen te leren en deze vaardigheden te gebruiken om werkende oplossingen te programmeren en te bouwen, en om natuurwetenschappelijke, technologische, technische en wiskundige (STEM) concepten met elkaar te verbinden.