

**Een overzicht van het
Programmeeronderwijs op Middelbare
scholen in Nederland**

Onderzoek van Onderwijs

Femke van Geel

Juli 2008

Begeleiders:

**Nico van Diepen
Nellie Verhoef**

Universiteit Twente

Samenvatting

Aangezien er geen standaard is voor Informatica onderwijs in Nederland varieert wat er wordt gegeven binnen dit vak van docent tot docent. Tijdens mijn diverse stageplekken (5 verschillende scholen en docenten), werd dit verschil voor mij met name tijdens de programmeerlessen erg duidelijk. In een poging deze verschillen enigszins in kaart te brengen, is er in opdracht van de Universiteit Twente een onderzoek gestart. Dit onderzoek was specifiek gericht op het programmeeronderdeel van het vak Informatica op het voortgezet onderwijs en werd uitgevoerd in de vorm van een enquête. De enquête heeft van 07-05-2008 tot en met 23-07-2008 online gestaan en was vanuit de InformaticaVO site te benaderen [1]. Er waren 24 respondenten.

Uit de resultaten van dit onderzoek wordt duidelijk wat het ontbreken van een norm of standaard voor een vak kan betekenen. Binnen alleen al het programmeeronderdeel voor het vak Informatica, zien we bijvoorbeeld immense verschillen qua urenverdeling: 10 lesuren ten opzichte van 140. Als meest populaire programmeertaal kiest men de opmaaktaal HTML. Zonder ook maar een enkele vorm van regels waaraan docenten zich moeten houden, is er blijkbaar een wildgroei ontstaan aan eigen opzetten voor het vak (of dan in ieder geval het programmeeronderdeel!) waarbij geen controle plaats vindt. De meest voor de hand liggende vorm van controle zou natuurlijk het invoeren van een centraal schriftelijk eindexamen (CSE) zijn. Maar in het vakdossier Informatica [4] wordt het extra duidelijk dat deze vorm van controle in elk geval niet is gewenst.

Inhoudsopgave

Introductie	4
1.1 Achtergrond	4
1.2 Onderzoeksdoel	4
1.3 Verslag overzicht	6
Theoretisch kader	7
2.1 Programmeeronderwijs, de standaard	7
2.2 Duits alternatief.....	8
Methode	11
3.1 Onderzoeksopzet.....	11
3.2 Uitvoering onderzoek	11
Resultaten	12
4.1 Verkregen data	12
4.1.1 Algemeen.....	12
4.1.2 Lesmethoden	12
4.1.3 Programmeeronderwijs	13
4.1.4 Docentenbeleving	13
4.1.5 Tijdsindeling.....	13
4.2 Interpretatie	14
Conclusies en aanbevelingen	16
5.1 Vervolg onderzoek	17
5.2 Naar aanleiding van het Duits onderzoek.....	18
Bijlagen	19
6.1 Appendix 1: enquête opzet.....	19
6.2 Appendix 2: ruwe data verkregen d.m.v. de enquête	22
Referenties	24

Hoofdstuk 1

Introductie

1.1 Achtergrond

In het schooljaar 1998-1999 werd in de bovenbouw van HAVO en VWO het keuzevak Informatica ingevoerd. Een nieuw vak, maar nog zonder docenten. Hiervoor werd de Consortium Omscholing Docenten Informatica (CODI) opgericht die actief is geweest met de scholing voor docenten Informatica. Inmiddels zijn we negen jaar verder: schooljaar 2007-2008. Ongeveer 60% van de scholen biedt Informatica aan. Op ongeveer 20% van deze scholen, wordt onbevoegd lesgegeven[1].

Informatica is een vak zonder Centraal Schriftelijk Eindexamen. Met andere woorden: eindexamen doet de leerling 'gewoon' op school. Hoewel er wel een handreiking is voor de eindtermen van het vak Informatica vanuit de Stichting Leerplan Ontwikkeling (SLO), is het slechts een richtlijn en geen regels die hier geboden wordt. De enige echte regel is dat alle examenonderdelen aan bod moeten komen, hoe dit precies wordt ingevuld, is aan de docent zelf. Dat betekent dat elke docent - tot op zekere hoogte - zijn eigen invulling kan geven aan de inhoud van het vak.

Zo kan het bijvoorbeeld voorkomen dat de docent A voor het onderdeel Programmeren zijn of haar leerlingen een website laat maken met behulp van Frontpage[2]. Docent B laat zijn of haar leerlingen bijvoorbeeld een dating service op het internet opzetten, inclusief het hosten van de dynamische webpagina, het maken, vullen en beheren van de database en wellicht nog een stand-alone programma hiervoor laten schrijven in C++.

Hieruit kunnen we afleiden dat het niveau van het Informatica onderwijs en specifiek het programmeeronderwijs, nogal verschilt op scholen. Dit onderzoek is er dan ook op gericht om deze verschillen in kaart te brengen.

1.2 Onderzoeksdoel

Om het niveau van programmeeronderwijs in kaart te kunnen brengen, zullen we eerst een afbakening van het onderzoeksgebied moeten vaststellen.

Er wordt specifiek gekeken naar het niveau van het programmeeronderwijs op scholen waar bevoegd Informatica wordt gegeven. Omdat het niet mogelijk is binnen beschikbare tijdsduur om alle scholen in Nederland waarop bevoegd wordt lesgegeven te onderzoeken, wordt ook hierin een selectie gemaakt (zie ook hoofdstuk 2).

Wanneer we deze afbakening gemaakt hebben, dienen we duidelijk te hebben wat precies wordt onderzocht. In dit geval dus het niveau van het programmeeronderwijs in Nederland. Deze vraag kunnen we niet in één klap beantwoorden. Immers, hoe bepalen we bijvoorbeeld het niveau? Om die vraag te kunnen beantwoorden moeten

we bepalen of we niveaus van programmeeronderwijs gaan vergelijken met een standaard of met elkaar.

Nu heeft het weinig zin om scholen qua programmeeronderwijs ten opzichte van elkaar te gaan vergelijken, dat zegt immers alleen iets van de situatie van het niveau van een school ten opzichte van andere scholen. Het liefst willen we vergelijken aan de hand van een standaard. Daarom wordt de hoofdonderzoeksvraag dan ook als volgt:

A: "Hoe(veel) verschilt het programmeeronderwijs bij het vak Informatica op middelbare scholen, gegeven door een bevoegde docent, ten opzichte van een standaard?"

Om hierop antwoord te kunnen geven, worden de volgende deelvragen gesteld:

B: "Wat is een standaard m.b.t. programmeeronderwijs?"

C: "Welk programmeeronderwijs wordt op de geselecteerde scholen gegeven?"

D: "Wat is het niveau van het programmeeronderwijs op de geselecteerde scholen?"

Door het beantwoorden van de deelvragen B, C en D, verkrijgen we voldoende informatie om ook hoofdonderzoeksvraag A te kunnen beantwoorden.

Om deze eerste deelvraag te kunnen beantwoorden, is een onderzoeksgroep nodig. Deze groep wordt geselecteerd uit de doelgroep van dit onderzoek: alle middelbare scholen die Informaticaonderwijs aanbieden. Om deze selectie te maken is een vooronderzoek nodig aan de hand van een te ontwerpen vragenlijst.

Door middel van deze vragenlijst moeten de volgende elementen bekend worden:

- Welke lesmethoden worden gebruikt voor het programmeeronderwijs
- Welke vorm van programmeeronderwijs worden er gegeven
 - Welke programmeertalen
 - Op welke manier
 - Zelfstudie
 - Klassikaal
 - Met een lesmethode
- Op welke manier wordt getoetst of de kennis voor het programmeeronderdeel correct is overgekomen
 - Toets
 - Op papier
 - Op de computer
 - Praktisch
 - Werkstuk wellicht i.c.m. schrijven van een programma
- Wat vinden de docenten programmeeronderwijs van niveau?

Om de tweede deelvraag te kunnen beantwoorden is een graadmeter nodig (derde deelvraag) om aan de hand daarvan het niveau te bepalen van het Informaticaonderwijs, specifiek voor het onderdeel programmeren. De handreiking van het SLO biedt (zoals duidelijk mag zijn) een handreiking maar geen vaste regels en invulling waaraan een en ander moet voldoen. Hieraan kan dus niet getoetst worden.

1.3 Verslag overzicht

Om te kunnen bepalen in hoeverre het programmeeronderwijs binnen het vak Informatica afwijkt van een standaard, zullen we eerst moeten bepalen wat deze standaard dan precies is. Dat dit niet zo gemakkelijk is als het ogenschijnlijk lijkt, zien we in hoofdstuk twee.

Zodra helder is wat we als de standaard beschouwen, kunnen we beginnen met het onderzoek en de opzet hiervan. In het derde hoofdstuk wordt deze onderzoeksopzet beschreven en gaan we in op de uitvoering van het onderzoek.

De resultaten van het onderzoek worden uitgebreid gepresenteerd in het vierde hoofdstuk. Elk van de onderzoeksvragen komt hier aan de orde en de data wordt gepresenteerd. In het vijfde hoofdstuk verbinden we dan conclusies aan deze data. Ook worden in dit hoofdstuk nog enkele aanbevelingen gedaan die nog leiden tot diverse mogelijkheden voor eventuele vervolgonderzoeken.

In hoofdstuk zes vindt u de bijlagen.

Hoofdstuk 2

Theoretisch kader

2.1 Programmeeronderwijs, de standaard

Op dit moment is er geen standaard beschikbaar wat betreft het vak Informatica in zijn algemeen en dus helaas ook niet voor het programmeeronderwijs dat binnen dit vak valt. De SLO Enschede heeft, zoals eerder vermeld, wel een handreiking met richtlijnen maar geen regels of standaard ter beschikking.

Dit houdt in dat een standaard zelf opgesteld moet worden. Daarmee wordt deelvraag B direct beantwoordt: er is geen standaard.

In dit hoofdstuk wordt aan de hand van de handreiking van het SLO en het beschikbare lesmateriaal voor programmeeronderwijs een standaard opgesteld. Aan de hand van deze standaard, zal de data die verkregen wordt het met onderzoek, worden vergeleken.

Voor Subdomein B3 (Software) staat 40 uur voor de kern ingedeeld. Subdomein B3 houdt het volgende in:

“De kandidaat beheerst eenvoudige datatypen, programmeerinstructies en programmeertechnieken.”

Het VWO kan nog een verdieping doen van 30 of 60 uren. De handreiking zegt dat de gebruikte programmeertaal deze vier punten moet ondersteunen:

- een connectie leggen met een database
- databaseinstructies kunnen opmaken en aanbieden aan het databasemanagementsysteem
- het resultaat van een selectie-opdracht op de database kunnen verwerken. Hiertoe beschikt de programmeertaal onder andere over herhalings- en keuzemogelijkheden
- de opmaak van een (dynamische) webpagina met in- en uitvoermogelijkheden verzorgen.

Hierbij wordt nog aangegeven dat HTML volstaat voor de opmaak van een webpagina, maar gezien HTML geen dynamische elementen kan verzorgen, hier dus een scriptingtaal gebruikt zou kunnen worden.

Naast deze vier punten, moeten leerlingen ook bekend zijn met Object georiënteerd programmeren, in die zin dat ze bepaalde OO begrippen begrijpen. Het VWO kan een verdiepingmodule doen waarin ze daadwerkelijk programmeren met een OO taal.

De items hierboven, geven duidelijke beperkingen aan de programmeertalen die een docent zou moeten kiezen. Voor de HAVO mag het sowieso duidelijk zijn dat hier

geen OO taal gekozen hoeft te worden. Voor het VWO is dat gelijk, tenzij de verdiepingsmodule in dit onderdeel valt.

Gezien de urenverdeling (40 HAVO/VWO, 30-60 uitbreiding VWO) en dat hierin eindterm 7 en subdomein B3 voldaan moet worden, zouden we een richtlijn kunnen aanhouden van 35 uren aan programmeeronderwijs en 5 uren aan theorie onderdelen die niet specifiek bij het programmeeronderdeel horen die de docent kiest (bijvoorbeeld aan OO begrippen als een docent ervoor kiest om geen OO taal te gebruiken). Binnen die 35 uren is de docent vrij om te kiezen welke programmeertaal of –talen hij/zij wilt gebruiken, zolang ze maar voldoen aan de items die hierboven zijn genoemd.

Een ideaal voorbeeld zou zijn:

Het maken van een dynamische website (met HTML/PHP) waarbij een leerling zijn eigen CD collectie bij kan houden. Er wordt dus een database gemaakt waar de informatie van de CD's instaan en vanuit de website moet een connectie naar de database gemaakt worden die in elk geval kan opvragen welke CD's er in de database staan. Het is aan de docent om aan te geven hoe ver hij hierin wilt doorgaan: toevoegen van nieuwe CD's, opvragen van CD's van een bepaald genre, CD's opvragen die in de categorie romantisch vallen maar niet 'love' in de titel hebben etc.

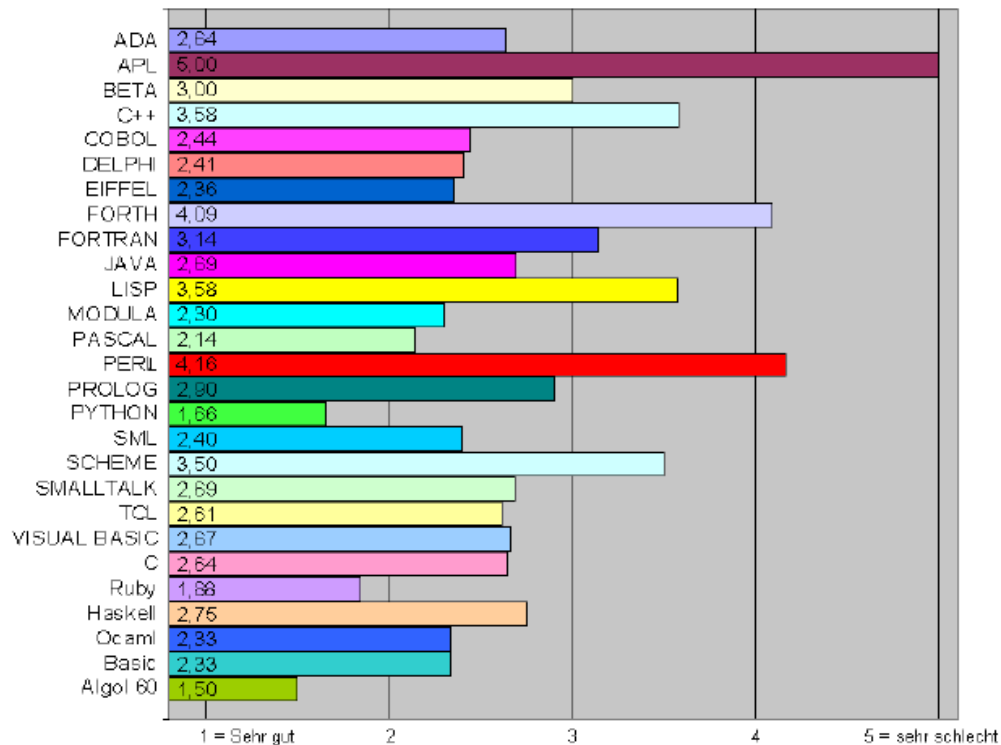
2.2 Duits alternatief

In Duitsland kent men een onderwijssysteem dat redelijk vergelijkbaar is met het systeem in Nederland. Ook hier wordt het vak Informatica aangeboden op middelbare scholen en betreft het een keuzevak. Ingo Linkweiler en Ludger Humbert [LH02] hebben een onderzoek gedaan naar programmeertalen die geschikt zijn voor secondary education. Zij stelden hiervoor diverse criteria op die wellicht ook zeer interessant zijn voor het Nederlandse vak Informatica (overzicht in het Duits):

Kriterien
Portabilität und Dateizugriff
Schneller GUI-Entwurf
Geeignete Hilfsmittel und Werkzeuge
Einfache Syntax und Semantik
Modularisierung und Wiederverwendbarkeit
Dokumentation
Softwaretest
Fehlersuche und Vermeidung
Nebenläufigkeit (Threading)
Erweiterungsfähigkeit und Einbettung
Automatische Speicherverwaltung
Objektorientierte Entwicklung
Funktionale Entwicklung
Prädikative Entwicklung
Einfache Lesbarkeit und Erlernbarkeit
Austausch von Algorithmen
Lernumgebung
Motivation

Figuur 1: overzicht criteria programmeertaal [LH02].

Met de data uit figuur 1 in het achterhoofd, hebben zij een onderzoek gedaan naar de 'leesbaarheid' van diverse programmeertalen. Dit onderzoek is uitgevoerd onder docenten Informatica. Dit leidde tot het volgende overzicht:



Figuur 2: overzicht leesbaarheid programmeertalen [LH02].

Het valt op dat er kennelijk nogal grote verschillen zijn in leesbaarheid van programmeertalen. Wat ook opvalt, is dat HTML niet in dit overzicht is opgenomen.

Linkweiler en Humbert kiezen in dit geval voor de programmeertaal Python. Python voldoet volgens Linkweiler en Humbert aan de meeste gestelde eisen, het overzicht van de criteria uit figuur 1 toegepast op Python ziet er als volgt uit:

Kriterien	für informatische Bildung	Eignung von Python
Portabilität und Dateizugriff	✓	++
Schneller GUI-Entwurf	◦	++
Geeignete Hilfsmittel und Werkzeuge	✓	++
Einfache Syntax und Semantik	✓	++
Modularisierung und Wiederverwendbarkeit	✓	++
Dokumentation	✓	++
Softwaretest	✓	++
Fehlersuche und Vermeidung	✓	++
Nebenläufigkeit (Threading)	✓	+
Erweiterungsfähigkeit und Einbettung	×	++
Automatische Speicherverwaltung	✓	++
Objektorientierte Entwicklung	✓	++
Funktionale Entwicklung	✓	+
Prädikative Entwicklung	✓	-
Einfache Lesbarkeit und Erlernbarkeit	✓	++
Austausch von Algorithmen	✓	++
Lernumgebung	✓	+
Motivation	✓	++

✓ ist erforderlich, ◦ ist nützlich, × ist unwichtig

++ sehr gut geeignet, + gut geeignet, ◦ ist geeignet, - eingeschränkt geeignet, - nicht geeignet

Figuur 3: criteria figuur 1 toegepast op Python [LH02].

We zien dat Python aan de meeste eisen voldoet. De conclusie van Linkweiler en Humbert is dan ook voor de hand liggend, zij bevelen Python aan als programmeertaal voor het Informatica onderwijs.

Of er lesmateriaal beschikbaar is voor deze programmeertaal is een heel andere vraag. In de referenties van dit artikel zien we genoeg handleidingen voor Python die er echter duidelijk niet op zijn gericht om leerlingen van het voortgezet onderwijs te leren programmeren met Python. De enige referentie die een indicatie levert van mogelijkheden voor leerlingen is van 2002 en op dit moment niet meer beschikbaar [Bu02].

We kunnen dus enkel uitgaan van de data die gepresenteerd wordt in de figuren 1 tot en met 3. Hieruit lijkt Python een goed alternatief te zijn als programmeertaal voor het voortgezet onderwijs. Zeker wanneer uit de ervaring van de docenten, figuur 2, aangegeven wordt dat dit een makkelijkere taal is dan de in dit onderzoek veel genoemde programmeertalen Java, Visual Basic en Delphi (zie 4.1.3).

Hoofdstuk 3

Methode

3.1 Onderzoeksopzet

Om het niveau van het programmeeronderwijs vast te stellen op een school X, zal er nogal wat data verzameld moeten worden. In dit geval is dat gebeurd door middel van een questionnaire. Op deze manier is getracht een zo groot mogelijk publiek te bereiken en docenten zelf de mogelijkheid te bieden wel dan niet mee te werken met het onderzoek en een mogelijk vervolgonderzoek.

De questionnaire werd geplaatst op de informaticaVO website [3], dit is een docenten community waar docenten lesmateriaal kunnen uitwisselen, met elkaar in discussie kunnen gaan over het vak, de leerstof, het lesmateriaal, kortom alles waar een Informatica docent behoefte aan zou kunnen hebben. Door hierop een oproep om mee te werken aan de questionnaire en een verwijzing naar waar deze te vinden is te plaatsen, zou een groot publiek bereikt moeten worden.

De questionnaire werd gemaakt met behulp van professionele enquête software van Surveysoftware van Enquête Compagnie[4].

De questionnaire is te vinden in de bijlagen.

3.2 Uitvoering onderzoek

De enquête heeft op InformaticaVO.nl gestaan van 07-05-2008 tot en met 23-07-2008. Gedurende die tijd hadden geregistreerde bezoekers de mogelijkheid deze enquête in te vullen.

Gedurende deze periode hebben 24 personen de enquête ingevuld.

Hoofdstuk 4

Resultaten

4.1 Verkregen data

De door de enquête verkregen data wordt in dit hoofdstuk gepresenteerd. Aan de data zijn geen aanpassingen gemaakt. De ruwe data is beschikbaar in Excel format, CSV format en MHTML document. Aan het eind van dit hoofdstuk is een paragraaf met conclusies en interpretatie van de verkregen data. Houdt er rekening mee dat een deel van de interpretatie persoonlijk is en dus kan verschillen met uw eigen mening. De ruwe data wordt ook nog gepresenteerd in de bijlagen.

4.1.1 Algemeen

Van de 24 respondenten, waren 4 vrouwen. Opvallend is, dat het merendeel van de respondenten een relatief hoge leeftijd heeft: zeven respondenten zijn 55 of ouder, elf zijn 45-55, twee zijn 35-45 en vier zijn 25-35. Geen van de respondenten was jonger dan 25.

Van al deze respondenten hebben 16 personen hun bevoegdheid behaald door het doen van de CODI cursus, 4 personen geven onbevoegd les 4 personen hebben een andere vorm van 'bevoegdheid', namelijk afgestudeerd in de Informatica (2), een wiskunde lesbevoegdheid (1) en één persoon die nog bezig was zijn lesbevoegdheid te halen.

4.1.2 Lesmethoden

De lesmethode die het meeste wordt gebruikt door de respondenten is Fundament van Instruct (8) gevolgd door Enigma dat voorheen bekend was als Turing (7), Informatica Actief (6) en Informatica van Edu'Actief (4). Daarnaast gaven 11 personen aan ook nog zelfgemaakt materiaal te gebruiken (veelal naast de hiervoor genoemde methoden). Eén persoon gaf aan naast Informatica Actief en eigen lesmateriaal ook nog veel online cursussen te gebruiken en een andere persoon gaf nog aan naast het eigen lesmateriaal de toets-it van Instruct te gebruiken.

Voor het programmeeronderdeel lijkt dit weinig verschil op te leveren, 11 personen geven aan dat ze voor het programmeeronderdeel dezelfde lesmethode(n) gebruiken als voor de rest van het vak Informatica. De mensen die switchen, gebruiken voor reguliere lessen allemaal Informatica Actief en switchen voor de programmeerlessen naar programmeer modules van Edu'Actief (5). Docenten die in de reguliere lessen gebruik maken van Fundament maken voor de programmeerlessen gebruik van de programmeer modules die Fundament levert. 10 personen geven nog aan dat ze nog ander materiaal gebruiken (11) waarvan 7 personen eigen gemaakt materiaal gebruiken en de overigen de lego NQC module, Gamemaker/ C# module, boeken over programmeren en online materiaal gebruiken.

Wanneer gevraagd wordt of de docenten vinden dat er genoeg lesmateriaal beschikbaar is voor het vak Informatica, dan antwoorden 8 personen dat er voldoende is. Vijf personen vinden dat er onvoldoende materiaal beschikbaar is, 8 personen vinden het niveau van het materiaal dat beschikbaar is niet voldoende, 1 persoon vindt dat het lesmateriaal dat beschikbaar is niet uitmaakt want eigen lesmateriaal werkt toch het prettigst. Nog 1 persoon vindt dat er voor de doelgroep te weinig lesmateriaal beschikbaar is en 1 persoon heeft eigenlijk geen idee.

4.1.3 Programmeeronderwijs

De programmeertalen die worden gebruikt zijn HTML (17 personen), Java (15), PHP (11), Visual Basic (7), Delphi (5) en C# (1). Veel genoemde alternatieven zijn Logo (2), Gamemaker Language (5), CSS (1), NQC (2), Alice (1), Scratch (1) en Python (1).

Het programmeeronderwijs wordt met name gegeven als een mix van klassikale lessen (14) en zelfstandig werken van de leerlingen (23). Alternatieve lessituaties worden gevonden in de manieren van PO's, projecten en presentaties (2).

Om het programmeeronderdeel af te ronden, laten 9 docenten hun leerlingen een theoretische toets maken. 16 docenten laten hun leerlingen een praktische toets maken op de computer en 18 docenten laten hun leerlingen een PO maken als afronding van dit onderdeel. Opmerkelijk is dat de meeste docenten minstens twee van deze manieren van afronding kiezen. Er zijn slechts 4 docenten die enkel kiezen voor de praktische toets (3) of de PO (1).

4.1.4 Docentenbeleving

Wanneer we docenten vragen naar een inschatting van het niveau van het programmeeronderwijs binnen het vak Informatica, dan vinden 2 personen dit ver beneden wat van dit onderwijs verwacht mag worden. Acht personen vinden het enigszins beneden verwachtingswaarde, de overige 14 respondenten vinden het op goed peil.

Als we dan specifiek vragen naar de inschatting van het niveau binnen hun eigen lessen ten aanzien van het landelijk niveau, dan vindt 1 persoon dat ver beneden niveau. Drie personen vinden het enigszins beneden niveau, 14 personen vinden het op peil en de overige 6 personen vinden dat zij boven niveau lesgeven.

4.1.5 Tijdsindeling

Wanneer we vragen naar de hoeveelheid tijd er wordt besteed aan het programmeeronderdeel dan varieert dit nogal. Voor de duidelijkheid, er staat 280 uur voor het hele vak Informatica op het VWO en 240 uur voor het hele vak op de HAVO. Programmeren hoort thuis in Subdomein B3 (Software) in de handreiking voor Informatica. 40 uren (HAVO) en 60 uren (VWO) staan gereserveerd voor dit hele subdomein.

De resultaten die wij verkrijgen wijken hier nogal vanaf:

13 personen geven 30 of minder lessen aan het programmeeronderdeel. 6 docenten geven 50 tot 70 lessen aan het programmeeronderdeel, 1 persoon besteedt 140 lessen en 1 persoon 190 lessen. Voor de precieze verdelingen, verwijzen we u naar tabel 1.

Hoeveel uren besteedt u aan het programmeeronderdeel?
15
20
16
15-30
12
140
50
70
40-60 HAVO 60-90 VWO
24
12
30
20
60
10
25-30
25
60
25
60
45
190

Tabel 1: uren voor het programmeeronderdeel

Bij de algemene opmerkingen van de docenten, leren we nog dat programmeeronderwijs niet alleen binnen het vak Informatica wordt gegeven, maar kennelijk ook in de onderbouw. Daarnaast wordt aangegeven dat er geen goed Nederlandstalig lesmateriaal is voor programmeeronderwijs in het VO. Er was één docent die aangaf de enquête lastig te vinden, gezien de nieuwe 2^{de} fase nogal verschilt met de oude.

4.2 Interpretatie

Dit onderzoek heeft een enorme hoeveelheid aan data opgeleverd, waar dan ook veel conclusies uit kunnen worden getrokken. We richten ons op de conclusies die met betrekking tot het onderdeel programmeren binnen het vak Informatica het interessantste zijn.

Opmerkelijk is dat hoewel HTML staat voor ‘HyperText Makeup Language’ en het dus een opmaaktaal is (en géén programmeertaal), het overgrote merendeel wél HTML gebruikt als programmeertaal. Ik kwam dit tegen tijdens mijn stage en was toen geschokt, maar van mening dat dit landelijk heus anders zou zijn. Dat bleek nogal een illusie. Kennelijk vindt het overgrote merendeel HTML wel degelijk een programmeertaal en gebruikt deze dan ook als zodanig voor zijn/haar programmeeronderwijs.

Een andere, nogal opmerkelijke, conclusie kunnen we trekken uit de resultaten op de vragen naar aanleiding van het niveau van het programmeeronderwijs in zijn algemeen en specifiek bij de ondervraagde in de klas. Het valt nogal op dat hoewel het merendeel van de ondervraagden aangeeft het niveau in het algemeen goed dan wel te laag vindt, dat kennelijk in zijn/haar eigen lessen probeert recht te trekken voor het landelijk gemiddelde. Het merendeel geeft hier aan dat het niveau goed of zelfs boven niveau is.

Wat ik persoonlijk het meest opmerkelijk vond, waren de resultaten naar aanleiding van de vraag betreffende de lesuren die besteed worden aan het programmeeronderdeel (zie tabel 1). Uiteraard kan men verwachten hier verschillen in te zien, maar de resultaten laten ons verschillen zien die immens uiteenlopen: van 10 lesuren tot meer dan een heel jaar aan programmeeronderwijs (190 lesuren). De vragen die we onszelf moeten stellen zijn dan ook: Hoe kan dit? Hoe kunnen we dit corrigeren?

Het antwoord op deze vragen zijn relatief simpel. Er is geen vastgestelde norm waaraan iedereen moet voldoen voor het vak Informatica. Er is een handreiking (dit is iets anders dan een handleiding) en een examenprogramma. Een docent die 10 lesuren besteedt aan programmeeronderwijs komt hier net zo goed mee weg als een docent die 190 lesuren programmeeronderwijs besteedt, even in het midden latend wat nu wenselijk is. Een oplossing hiervoor zou zijn het instellen van vaste regels waaraan docenten zich binnen dat vak aan moeten houden. Dit ervoor moeten zorgen dat de cijfers landelijk wat gelijkjer worden. Dat zou dan getoetst kunnen worden op landelijk niveau. Dit klinkt al heel erg als een centraal examen. Dat zou dan ook een oplossing kunnen zijn voor de verschillen die we hier zien binnen het vak Informatica. Probleem hierbij is dat de docenten hier niets voor lijken te voelen. In het vakdossier voor Informatica [4], wordt duidelijk dat het overgrote merendeel van de docenten géén centraal schriftelijk eindexamen wenst:

<i>Antwoordcategorie</i>	<i>Score</i>
Een centraal examen is voor mij onbespreekbaar	29.2%
Een centraal examen is voor mij slechts onder voorwaarden bespreekbaar	26.1%
Ik ben geen voorstander van een centraal examen, maar heb er geen bezwaar tegen	16.9%
Ik zie wel voordelen van een centraal examen	13.8%
Ik ben warm voorstander van een centraal examen	13.8%

Tabel 2: oordeel van sectieleiders op een eventuele instelling van een central examen [4].

Wanneer we naar de cijfers in tabel 2 kijken, lijkt een centraal examen voorlopig nog geen optie. Gezien we verder enkel de handreiking hebben van het SLO en er geen vorm van controle plaatsvindt, zal er voorlopig waarschijnlijk weinig veranderen binnen het programmeeronderdeel van het vak Informatica. Het is aan de docenten zelf om na te gaan of een urenverdeling van 190 lesuren programmeeronderwijs nu echt wel wenselijk is of niet.

Hoofdstuk 5

Conclusies en aanbevelingen

Aan de hand van de verkregen data kunnen we inmiddels de onderzoeksvragen beantwoorden. Wanneer we uitgaan van de gestelde standaard in Hoofdstuk 2, kunnen we zien dat qua tijdsindeling, het merendeel van de docenten redelijk aardig in de buurt komt van de 35 uur. Wanneer we in ons achterhoofd houden dat een derde (8) van de docenten redelijk in de buurt blijft van de 35 uren programmeren (25-45 uren is acceptabel) dan rijst onmiddellijk de vraag wat de docenten doen die zich hier niet aan houden. Acht van de vierentwintig ondervraagden zit onder de 25 uren en acht zitten daarboven. Van de acht die eronder zitten, vragen we ons af wat zij met de tijd doen die ze dan overhouden. Van de acht die erboven zitten, vragen we ons af wat moet wijken voor de programmeerlessen.

Wanneer we ervan uitgaan dat de 24 respondenten van dit onderzoek een goede afspiegeling van de werkelijkheid zijn, zouden we kunnen concluderen dat ongeveer een derde van alle docenten zich qua tijd goed aan de standaard houdt.

Richten we ons op de inhoud van het programmeeronderdeel dan valt op dat veel docenten HTML gebruiken als programmeertaal. HTML op zich is niet afdoende, zoals dat in de standaard is gesteld. Er moet naast HTML nog een (scripting)taal gebruikt worden om een webpagina dynamisch te maken. Het zou mogelijk kunnen zijn dat de 15 personen die hebben aangegeven Java te gebruiken als programmeertaal, eigenlijk Javascript bedoelen, dit is slechts giswerk en hier is geen data van beschikbaar. PHP is daarentegen wel een goede aanvulling op HTML voor het dynamisch maken van een webpagina. PHP kan ook goed worden gebruikt om aan de andere voorwaarden (zie de vier punten in hoofdstuk 2.1) te voldoen. Ook Visual Basic is hiervoor geschikt. CSS wordt nog als programmeertaal aangegeven en ook hiervoor geldt dat het geen programmeertaal is. Het is een opmaakelement.

Wat betreft of de docenten die gereageerd hebben op deze enquête zich dus houden aan de door ons gestelde standaard is nog maar de vraag. Zeker omdat veel personen Java hebben gekozen als programmeertaal, terwijl dit een erg lastige OO-taal is om te leren, zeker binnen de limiet van 35 uur. Mochten deze mensen allemaal JavaScript hebben bedoeld, dan is er zeker hoop. In combinatie met HTML (of HTML in combinatie met PHP) zijn we zeker op goede weg. Docenten die echter maar 1 programmeertaal kiezen, kunnen haast onmogelijk aan alle gestelde eisen van 2.1 voldoen. Een docent zal dus altijd minimaal 2 programmeertalen moeten kiezen.

Wat verder ook opvalt is de lage dunk die de respondenten van dit onderzoek lijken te hebben van het niveau van het vak Informatica. Wanneer we kijken naar de antwoorden in het algemene niveau, dan zien we dat het merendeel rond 'goed' of lager zit. Waarom deze personen naar een lager niveau neigen is onduidelijk: zelf vinden ze van zichzelf toch dat ze het goed doen en neigen ze hier zelfs naar boven gemiddeld. Kennelijk vindt men zijn collega's toch van lager niveau dan het eigen niveau. Redenen hiervoor zijn niet duidelijk. Gezien er geen vorm van controle is op het onderwijs dat wordt gegeven (geen CSE, dus ook geen vergelijk in resultaten per school), is het

vreemd dat men blijkbaar de collega's lager inschat, gezien er haast op geen enkele manier inzicht is in wat die collega's dan doen.

Al deze conclusies en die van hoofdstuk 3, leiden tot een aantal opzetten voor een vervolg onderzoek. Deze worden besproken in de volgende paragraaf.

5.1 Vervolg onderzoek

Wanneer we terugkijken naar onze onderzoeksvragen uit het eerste hoofdstuk, dan zijn al de deelvragen inmiddels wel beantwoord. De enige die ons nog problemen geeft, is de hoofdvraag:

A: "Hoe(veel) verschilt het programmeeronderwijs bij het vak Informatica op middelbare scholen, gegeven door een bevoegde docent, ten opzichte van een standaard?"

Wanneer we kunnen uitgaan van een goede afspiegeling van de werkelijkheid wat betreft onze groep respondenten, dan kunnen we in elk geval zeggen dat qua tijdsindeling ongeveer een derde van de docenten zich hier naar alle redelijkheid aan houdt. Dit houdt in dat tweederde dat dus niet doet. Van deze tweederde zit de helft boven en de helft onder deze standaard. Wat docenten doen met de tijd die ze over hebben, of waar ze de extra tijd vandaan halen, is niet bekend.

Daarom raden we in eerste instantie aan om nog een onderzoek uit te voeren dat specifiek is gericht op de verdeling van de uren binnen het gehele vak Informatica zodat we concrete data hebben hierover. Dan kunnen we wellicht ook aangeven waar extra tijd vandaan moet worden gehaald voor het programmeeronderdeel of waar we tijd van het programmeren weg kunnen halen om te reserveren voor andere onderdelen uit het examenprogramma. Het zou waardevol zijn om te weten in hoeverre docenten deze handreiking van het SLO (zowel qua examenprogramma als indeling) volgen en waar het significant anders gaat. Aan de hand van deze data zouden we belangrijke aanbevelingen kunnen doen betreffende de inhoud en tijdsindeling van het vak Informatica.

Inhoudelijk gezien staan er een aantal eisen in de standaard (uit de handreiking van het SLO) waar het programmeeronderdeel aan moet voldoen. We laten hier de uitbreiding buiten beschouwing. Wanneer we kritisch kijken naar de antwoorden gegeven in de enquête, moeten we concluderen dat een groot deel van de docenten hier zich niet aan houdt en ook niet in de buurt komt. HTML wordt veel gebruikt, dit is de populairste programmeertaal. PHP wordt ook gebruikt maar in veel mindere mate; wanneer we er dan maar vanuit gaan dat dit in combinatie met HTML is, komt in elk geval dit selecte groepje in de buurt van de standaard (in elk geval qua mogelijkheden). Het is natuurlijk nog maar de vraag of docenten hun leerlingen ook deze mogelijkheden laten benutten (of verplichten te benutten). Strikt gesproken in de enquête staat Java met stip op nummer 2 als programmeertaal die veel gebruikt wordt. Java (en niet Javascript) is geen uitbreiding op HTML en mag dus feitelijk niet zo worden gezien. Echter, het lijkt ons sterk dat een groot deel van de docenten hun leerlingen Java probeert te leren, zeker gezien het feit dat elke docent minimaal twee programmeertalen leert. Zelfs met een erg hoog aantal lessen van 140, zou dat betekenen dat er ruwweg 70 in Java zouden gaan en dat is erg weinig voor een OO programmeertaal als Java. Mochten we er vanuit gaan

dat elke docent die Java koos eigenlijk Javascript bedoelde, dan is het beeld iets minder somber. Helaas is dit niet naar voren gekomen binnen deze enquête.

Als tweede aanbeveling willen we dan ook een onderzoek naar specifiek de programmeertalen die worden gebruikt door docenten om de standaard te halen (indien dat het geval is). Uiteraard is het interessant om te weten wie van de respondenten daadwerkelijk Java bedoelde en wie eigenlijk JavaScript bedoelde. Mocht er daadwerkelijk onderscheid zijn in deze groep mensen die toch allemaal Java kozen, dan kunnen we ons afvragen waarom een docent Informatica het verschil tussen Java en JavaScript niet weet.

Als laatste aanbeveling voor het vervolg onderzoek willen we wijzen naar een onderdeel dat eigenlijk niet binnen dit onderzoek naar voren is gekomen maar desalniettemin zeer belangrijk is: het CSE. Het SLO heeft met onderzoeken al laten zien dat de gemiddelde docenten Informatica hier niets in ziet. De hoofdreden was: omdat docenten dan de ruimte binnen het vak kwijtraken (de 2^{de} reden was omdat Informatica bij geen vervolg studie als verplicht staat aangemerkt). Het zou waardevol zijn om zonder een CSE toch een mate van controle en daaraan gelinked dus een standaard te hebben. Oplossingen hiervoor zouden in een nieuw onderzoek samen met docenten kunnen worden bedacht. Hiermee zouden de belangrijkste bezwaarhebbers hun eigen invloed kunnen aanwenden om de kwaliteit van hun eigen vak te verbeteren. Dat is uiteindelijk iets dat elke docent voor ogen zou moeten hebben.

5.2 Naar aanleiding van het Duits onderzoek

Het onderzoek zoals dat in Duitsland heeft plaatsgevonden, geeft een aardige opzet van criteria waaraan een programmeertaal zou moeten voldoen in het voortgezet onderwijs. In hun paper nemen Humbert en Linkweiler de programmeertaal Python, als ideale onderwijsprogrammeertaal. Het onderzoek is in Duitland uitgevoerd en hoewel het onderwijssysteem redelijk hetzelfde lijkt, is er niet veel bekend over het informaticaonderwijs binnen het voortgezet onderwijs. Om dus zonder meer de programmeertaal Python aan te bevelen in de Nederlandse setting is dus wat vergaand. Daarom is een verder onderzoek naar de inzetbaarheid van de programmeertaal Python in het Nederlandse voortgezet onderwijssysteem binnen het vak Informatica zeker aan te bevelen.

Afhankelijk van de resultaten van een zo'n onderzoek, zouden zelfs stappen kunnen worden ondernomen om deze programmeertaal als verplicht te stellen binnen het vak Informatica. Dit geeft docenten nog steeds een zekere mate van vrijheid; ze zijn vrij om zelf de tweede programmeertaal te kiezen. Mocht Python inderdaad zo veelbelovend blijken als het in het onderzoek van Humbert en Linkweiler wordt gepresenteerd dan heeft een verplichtstelling van Python in elk geval tot resultaat dat leerlingen zich de basisprincipes van programmeren redelijk eenvoudig eigen kunnen maken. Dit zou dan ten goede komen van het niveau van het programmeeronderwijs binnen het vak Informatica. Wellicht dat Nederlandse Informaticadocenten hier een geschikt alternatief voor de opmaaktaal HTML in kunnen vinden.

Hoofdstuk 6

Bijlagen

6.1 Appendix 1: enquête opzet.

Programmeeronderwijs op middelbare scholen

1.	Bent u een man of een vrouw?
	<input type="radio"/> Man <input type="radio"/> Vrouw
2.	In welke leeftijdscategorie valt u?
	<input type="radio"/> Jonger dan 25 <input type="radio"/> 25 tot 35 <input type="radio"/> 35 tot 45 <input type="radio"/> 45 tot 55 <input type="radio"/> Ouder dan 55
3.	Heeft u een lesbevoegdheid voor het vak Informatica?
	<input type="radio"/> Ja, ik heb de CODI cursus gedaan en succesvol afgerond. <input type="radio"/> Nee, ik ben onbevoegd. <input type="radio"/> Anders, namelijk... <input type="text"/>
4.	Welke lesmethode(n) gebruikt u in het algemeen? (meerdere antwoorden mogelijk)
	<input type="checkbox"/> Fundament van Instruct <input type="checkbox"/> Enigma (voorheen Turing) <input type="checkbox"/> Informatica van Edu'Actief <input type="checkbox"/> Informatica Actief <input type="checkbox"/> Ik gebruik zelfgemaakt materiaal <input type="checkbox"/> Anders, namelijk... <input type="text"/>

5.	Welke lesmethode(n) gebruikt u voor het programmeeronderwijs dat u geeft? (meerdere antwoorden mogelijk)
	<input type="checkbox"/> Programmeermodules van Instruct <input type="checkbox"/> Programmeermodules van Edu'Actief Informatica <input type="checkbox"/> Reguliere methode van vraag 4. <input type="checkbox"/> Anders, namelijk... <input type="text"/>

6.	Welke programmeertaal/-talen behandelt u? (meerdere antwoorden mogelijk)
	<input type="checkbox"/> Pascal <input type="checkbox"/> Delphi <input type="checkbox"/> Java <input type="checkbox"/> HTML <input type="checkbox"/> C sharp <input type="checkbox"/> C++ <input type="checkbox"/> PHP <input type="checkbox"/> Visual Basic <input type="checkbox"/> Anders, namelijk... <input type="text"/>

7.	Op welke manier onderwijst u het programmeeronderwijs? (meerdere antwoorden mogelijk)
	<input type="checkbox"/> Klassikale lessen <input type="checkbox"/> Zelfstandig werken van de leerlingen <input type="checkbox"/> Anders, namelijk <input type="text"/>

8.	Hoe rondt u het programmeeronderdeel af? (meerdere antwoorden mogelijk)
	<input type="checkbox"/> Theoretische toets (op papier) <input type="checkbox"/> Praktische toets (op de computer) <input type="checkbox"/> Praktische Optracht <input type="checkbox"/> Anders, namelijk... <input type="text"/>

9.	Wat denkt u dat het niveau is van programmeeronderwijs op de middelbare scholen?
	<input type="radio"/> Ver beneden het niveau <input type="radio"/> Enigszins beneden niveau <input type="radio"/> Op goed niveau <input type="radio"/> Boven het niveau <input type="radio"/> Ver boven het niveau

10.	Wat vindt u van het niveau van het programmeeronderwijs dat u zelf geeft?
	<input type="radio"/> Ver beneden niveau <input type="radio"/> Enigszins beneden niveau <input type="radio"/> Op goed niveau <input type="radio"/> Boven niveau <input type="radio"/> Ver boven niveau

11.	Vindt u dat er genoeg goed lesmateriaal beschikbaar is voor het programmeeronderdeel?
	<input type="radio"/> Ja, er is zoveel dat ik door de bomen het bos niet meer zie <input type="radio"/> Ja, voldoende <input type="radio"/> Nee, er is onvoldoende <input type="radio"/> Nee, er is onvoldoende lesmateriaal op niveau (er is wel veel lesmateriaal) <input type="radio"/> Anders, namelijk... <input type="text"/>

12.	Hoeveel tijd besteed u aan het programmeeronderdeel?
	<input type="text" value="...lesuren"/>

13.	Heeft u nog opmerkingen of vragen? (Gaarne uw emailadres invullen als u een reactie verwacht).
	<input type="text" value="<emailadres>"/>

6.2 Appendix 2: ruwe data verkregen d.m.v. de enquête

Bent u een man of een vrouw?		
Man	#20	83%
Vrouw	#4	17%
<i>n</i> =24		

In welke leeftijdscategorie valt u?		
Jonger dan 25	#0	0%
25 tot 35	#4	16%
35 tot 45	#2	8%
45 tot 55	#12	48%
Ouder dan 55	#7	28%
<i>n</i> =25		

Heeft u een lesbevoegdheid voor het vak Informatica?		
Ja, ik heb de CODI cursus gedaan en succes...	#17	68%
Nee, ik ben onbevoegd.	#4	16%
Anders, namelijk...	#4	16%
<i>n</i> =25		

Welke lesmethode(n) gebruikt u in het algemeen? (meerdere antwoorden mogelijk)		
Fundament van Instruct	#8	19%
Enigma (voorheen Turing)	#7	17%
Informatica van Edu'Actief	#5	12%
Informatica Actief	#7	17%
Ik gebruik zelfgemaakt materiaal	#11	26%
Anders, namelijk...	#2	5%
<i>n</i> =42		

Welke lesmethode(n) gebruikt u voor het programmeeronderwijs dat u geeft? (meerdere antwoorden mogelijk)		
Programmeermodules van Instruct	#7	16%
Programmeermodules van Edu'Actief Informat...	#6	14%
Reguliere methode van vraag 4.	#11	25%
Anders, namelijk...	#10	23%
<i>n</i> =44		

Welke programmeertaal/-talen behandelt u? (meerdere antwoorden mogelijk)		
Pascal	#3	4%
Delphi	#5	7%
Java	#15	21%
HTML	#18	25%
C sharp	#1	1%
C++	#0	0%
PHP	#11	15%
Visual Basic	#8	11%
Anders, namelijk...	#3	4%
<i>n</i> =72		

Op welke manier onderwijst u het programmeeronderwijs? (meerdere antwoorden mogelijk)		
Klassikale lessen	#14	33%
Zelfstandig werken van de leerlingen	#24	57%
Anders, namelijk	#2	5%
<i>n</i> =42		

Hoe rondt u het programmeeronderdeel af? (meerdere antwoorden mogelijk)		
Theoretische toets (op papier)	#9	20%
Praktische toets (op de computer)	#17	38%
Praktische Opdracht	#19	42%
Anders, namelijk...	#0	0%

n=45

Kunt u een inschatting geven van het niveau van het programmeeronderwijs binnen het vak Informatica?		
Ver beneden het niveau	#2	8%
Enigszins beneden niveau	#8	33%
Op goed niveau	#14	58%
Boven het niveau	#0	0%
Ver boven het niveau	#0	0%

n=24

Kunt u een inschatting maken van het niveau programmeeronderwijs dat u geeft ten aanzien van het landelijke niveau?		
Ver beneden niveau	#1	4%
Enigszins beneden niveau	#4	16%
Op goed niveau	#14	56%
Boven niveau	#6	24%
Ver boven niveau	#0	0%

n=25

Vindt u dat er genoeg goed lesmateriaal beschikbaar is voor het programmeeronderdeel?		
Ja, er is zoveel dat ik door de bomen het ...	#0	0%
Ja, voldoende	#8	32%
Nee, er is onvoldoende	#6	24%
Nee, er is onvoldoende lesmateriaal op n...	#8	32%
Anders, namelijk...	#3	12%

n=25

De data voor de open vragen wordt gepresenteerd in hoofdstuk 4.

Hoofdstuk 7

Referenties

[1] Informatie van de CODI commissie, te vinden op de Informatica VO site:
<http://www.informaticavo.nl>

[2] Frontpage is een onderdeel van MS Office. <http://office.microsoft.com/nl-nl/frontpage/FX100647001043.aspx>

[3] InformaticaVO website is te vinden op <http://www.informaticaVO.nl>

[4] EnqueteCompagnie website is te vinden op <http://www.enquetecompagnie.com>

Handreiking Informatica (SLO) te vinden via:

http://www.slo.nl/themas/00108/Map2/Handreiking_Informatica_DEFINITIEF.pdf

Examenprogramma Informatica (SLO) te vinden via:

http://www.slo.nl/themas/00108/Map2/Examenprogramma_informatica_DEFINITIEF.pdf

[Bu02] B. Burley. *Python Bibliotheca, Python resources for teachers and students*,
<http://www.ibiblio.org/obp/pyBiblio/interactive.php>, 06/2002

[LH02] I. Linkweiler, L. Humbert. *Ergebnisse der Untersuchung zur Eignung einer Programmiersprache für die schnelle Softwareentwicklung – kann der Informatikunterricht davon profitieren?*, Didaktik der Informatik, Universität Dortmund, 2002.