

Programmeren & Correctheid

Docent: Prof. dr. F.S. de Boer*

Assistenten: Michiel Helvensteijn[†] and Stijn de Gouw[‡]

Literatuur

Verification of Sequential and Concurrent Programs.

Krzysztof R. Apt, Frank S. de Boer, Ernst-Rüdiger Olderog.

Series: Texts in Computer Science. Springer.

3rd ed. 2nd Printing. ISBN: 978-1-84882-744-8.

Te behandelen stof

Course Towards Object-Oriented Program Verification (zie *Preface: Outlines of One-Semester Courses* en slides).

Agenda

04-2 While Programs: Syntax and Semantics (3.1 en 3.2).

Opgaven: 2.3 en 3.9(i).

Extra opgave: wat is de relatie tussen de (waarden) van de variabelen x , y , u and v van het volgende while-programma:

```
 $y := 0; u := 0; v := 1;$ 
```

```
while  $u + v \leq x$ 
```

```
do  $y := y + 1;$ 
```

```
     $u := u + v;$ 
```

```
     $v := v + 2$ 
```

```
od
```

11-2 While Programs: Semantics and Proof Theory (3.2 en 3.3).

Opgave 3.4. Extra opgaven:

*frb@cwi.nl

†michiel.helvensteijn@cwi.nl

‡cdegouw@liacs.nl

1. Bewijs met inductie naar de structuur van S dat als $\langle S, \sigma \rangle \rightarrow \langle S_1, \sigma_1 \rangle$ en $\langle S, \sigma \rangle \rightarrow \langle S_2, \sigma_2 \rangle$ dan $S_1 \equiv S_2$ en $\sigma_1 = \sigma_2$.
2. Geef de berekening voor $\langle z := x; x := y; y := z, \sigma \rangle$ en laat zien dat voor de eindtoestand σ' geldt dat $\sigma'(x) = \sigma(y)$ en $\sigma'(y) = \sigma(x)$.
3. Voor welke statements gelden de volgende correctheidsbeweringen.
 - (a) $\{\mathbf{true}\}S\{\mathbf{true}\}$
 - (b) $\{\mathbf{false}\}S\{p\}$
 - (c) $\{\mathbf{true}\}S\{\mathbf{false}\}$

18-2 While Programs: Proof Theory (3.3).

Opgaven: 2.2, 3.5, 3.9(ii) (in System PW), 3.10(i).

25-2 While Programs: Summation Problem (3.1) en Case Study: Minimum-Sum Section Problem (3.11).

Opgaven: Proeftentamen.

04-3 Recursive Programs: Syntax and Semantics (4.1 and 4.2).

Werkcollege: Bespreking proeftentamen.

11-3 Recursive Programs: Proof Theory (4.3).

Opgaven: 4.1(i). Extra opgaven:

1. Definieer de *maximale recursiediepte* van een berekening $\langle S, \sigma \rangle \rightarrow^* \langle E, \tau \rangle$ met inductie naar lengte van de berekening.
2. Laat a een array zijn van type $\mathbf{integer} \rightarrow \mathbf{integer}$, k en sum simpele variabelen van type $\mathbf{integer}$ en $S \equiv \mathbf{if } k \leq N \mathbf{ then } sum := a[k] + sum; k := k + 1; P \mathbf{ fi}$.
Bewijs

$$\{\mathbf{true}\} P :: S \mid k := 0; sum := 0; P \{sum = \sum_{i=0}^N a[i]\}$$

18-3 Extra oefeningen (slides 77 en 78).

25-3 Tool Support: Mist Introduction.

Werkcollege: Deeltentamen.

01-4 Bespreking deeltentamen.

Werkcollege: Prakticum Mist inleveropgave.

08-4 Case studies While programs (Minimum Search and Bubble Sort).

Werkcollege: Inleveropgave Selection Sort.

15-4 Recursive Programs with Parameters (5.1 en 5.2).

Extra opgaven:

- Laat zien hoe je recursieve programma's met parameters kunt modeleren met behulp van lokale variabelen (in een block statement).
- Geef de berekening van $Ct(2)$, voor de procedure Ct in Example 5.2.
- Laat zien hoe je een proof-outline van een while statement **while** B **do** S **od** kunt omzetten in een proof-outline voor de body **if** B **then** $S; P$ **fi** van de recursieve procedure P .

29-4 Recursive Programs with Parameters (5.3).

06-5 Total Correctness While Programs (3.3).

Tijdens het werkcollege wordt het “herdeeltentamen” gegeven.

Evaluatie

- De behandelde stof wordt schriftelijk geëxamineerd .
- Daarnaast wordt er één *deeltentamen* gegeven tijdens het werkcollege op 25-3. Op het laatste werkcollege van 06-5 is er een “herdeeltentamen”.
- Tot slot, moeten er nog 3 opgaven gemaakt en ingeleverd worden.

Voor het eindcijfer telt het schriftelijk tentamen voor 50%. Het deeltentamen en het maximum van de inleveropgaven tellen beide voor 25%. Zie uitslagen inleveropgaven, deeltentamens, tentamen en de einduitslag.