Inhoud

[Tabel met vereisten en welke daar van voltooid zijn 1](#_Toc53759937)

[Uitbreidingen aan de compiler 5](#_Toc53759938)

# Tabel met vereisten en welke daar van voltooid zijn

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Omschrijving | Prio | Punten | Competentie VT | Afgerond |
| AL01 | De code behoudt de packagestructuur van de aangeleverde startcode. Toegevoegde code bevindt zich in de relevante packages. | Must | 0 | APP-6 | |  | | --- | |  | |
| AL02 | Alle code compileert en is te bouwen met Maven 3.6 of hoger, onder OpenJDK 13. Tip: controleer dit door eerst mvn clean uit te voeren alvorens te compileren en in te leveren, hierop een onvoldoende halen is echt zonde. Gebruik van Oracle versies van Java is uitdrukkelijk niet toegestaan. | Must | 0 | n.v.t. | |  | | --- | |  | |
| AL03 | De code is goed geformatteerd, zo nodig voorzien van commentaar, correcte variabelenamen gebruikt, bevat geen onnodig ingewikkelde constructies en is zo onderhoudbaar mogelijk opgesteld. (naar oordeel van docent) | Must | 0 | n.v.t. | |  | | --- | |  | |
| AL04 | De docent heeft vastgesteld (tijdens les, assessment of op een andere manier) dat de compiler eigen werk is en dat je voldoet aan de beoordelingscriteria van APP-6, te weten: - Kent de standaardarchitectuur van compilers; - Kent de basisbegrippen over programmeertalen (zoals syntaxis, semantiek). | Must | 0 | APP-6 | |  | | --- | |  | |
| PA00 | De parser dient zinvol gebruik te maken van jouw eigen implementatie van een stack generic voor ASTNode (VT: zie huiswerk IHANStack<ASTNode>) | Must | 0 | APP-1, APP-9 | |  | | --- | |  | |
| PA01 | Implementeer een grammatica plus listener die AST’s kan maken voor ICSS documenten die “eenvoudige opmaak” kan parseren, zoals beschreven in de taalbeschrijving. In level0.icss vind je een voorbeeld van ICSS code die je moet kunnen parseren. testParseLevel0() slaagt. | Must | 10 | APP-6, APP-7 | |  | | --- | |  | |
| PA02 | Breid je grammatica en listener uit zodat nu ook assignments van variabelen en het gebruik ervan geparseerd kunnen worden. In level1.icss vind je voorbeeldcode die je nu zou moeten kunnen parseren. testParseLevel1() slaagt. | Must | 10 | APP-6, APP-7 | |  | | --- | |  | |
| PA03 | [Breid je grammatica en listener uit zodat je nu ook optellen en aftrekken en vermenigvuldigen kunt parseren. In level2.icss vind je voorbeeld- code die je nu ook zou moeten kunnen parseren. Houd hierbij rekening met de rekenregels (vermenigvuldigen gaat voor optellen en aftrekken, optellen en aftrekken gaan van links naar rechts; zie ook deze site.”testParseLevel2() slaagt.](https://www.beterrekenen.nl/website/index.php?pag=217) | Must | 10 | APP-6, APP-7 | |  | | --- | |  | |
| PA04 | Breid je grammatica en listener uit zodat je if/else-statements aankunt. In level3.icss vind je voorbeeldcode die je nu ook zou moeten kunnen parseren. testParseLevel3() slaagt. | Must | 10 | APP-6, APP-7 | |  | | --- | |  | |
| PA05 | PA01 t/m PA04 leveren minimaal 30 punten op | Must | 0 | nvt | |  | | --- | |  | |
| CH00 | Minimaal vier van onderstaande checks moeten zijn geïmplementeerd | Must | 0 | APP-2, APP-6, APP-7, | |  | | --- | |  | |
| CH01 | Controleer of er geen variabelen worden gebruikt die niet gedefinieerd zijn. | Should | 5 |  | |  | | --- | |  | |
| CH02 | Controleer of de operanden van de operaties plus en min van gelijk type zijn. Je mag geen pixels bij percentages optellen bijvoorbeeld. Controleer dat bij vermenigvuldigen minimaal een operand een scalaire waarde is. Zo mag 20% \* 3 en 4 \* 5 wel, maar mag 2px \* 3px niet. | Should | 5 |  | |  | | --- | |  | |
| CH03 | Controleer of er geen kleuren worden gebruikt in operaties (plus, min en keer). | Should | 5 |  | |  | | --- | |  | |
| CH04 | Controleer of bij declaraties het type van de value klopt met de property. Declaraties zoals width: #ff0000 of color: 12px zijn natuurlijk onzin. | Should | 5 |  | |  | | --- | |  | |
| CH05 | Controleer of de conditie bij een if-statement van het type boolean is (zowel bij een variabele-referentie als een boolean literal) | Should | 5 |  | |  | | --- | |  | |
| CH06 | Controleer of variabelen enkel binnen hun scope gebruikt worden | Must | 5 |  | |  | | --- | |  | |
| TR01 | Implementeer de EvalExpressions transformatie. Deze transformatie vervangt alle Expression knopen in de AST door een Literal knoop met de berekende waarde. | Must | 10 | APP-2, APP-6, APP-7 | |  | | --- | |  | |
| TR02 | Implementeer de RemoveIf transformatie. Deze transformatie verwijdert alle IfClauses uit de AST. Wanneer de conditie van de IfClause TRUE is wordt deze vervangen door de body van het if-statement. Als de conditie FALSE is dan vervang je de IfClause door de body van de ElseClause. Als er geen ElseClause is bij een negatieve conditie dan verwijder je de IfClause volledig uit de AST. | Must | 10 | APP-2, APP-6, APP-7 | |  | | --- | |  | |
| GE01 | Implementeer de generator in nl.han.ica.icss.generator.Generator die de AST naar een CSS2-compliant string omzet. | Must | 5 | APP-2, APP-6, APP-7 | |  | | --- | |  | |
| GE02 | Zorg dat de CSS met twee spaties inspringing per scopeniveau gegenereerd wordt. | Must | 5 | APP-2, APP-6, APP-7 | |  | | --- | |  | |

# Uitbreidingen aan de compiler

De volgende uitbreindingen zijn in goedgekeurd door de docent.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Omschrijving** | **MoSCoW** | **Aantal Punten** | **Afgerond** |
| Extra 1 | Boolean expressies + evaluatie | Could | 10 | Nee |
| Extra 2 | Dubbele variabele voorkomen | Could | 2 | Nee |