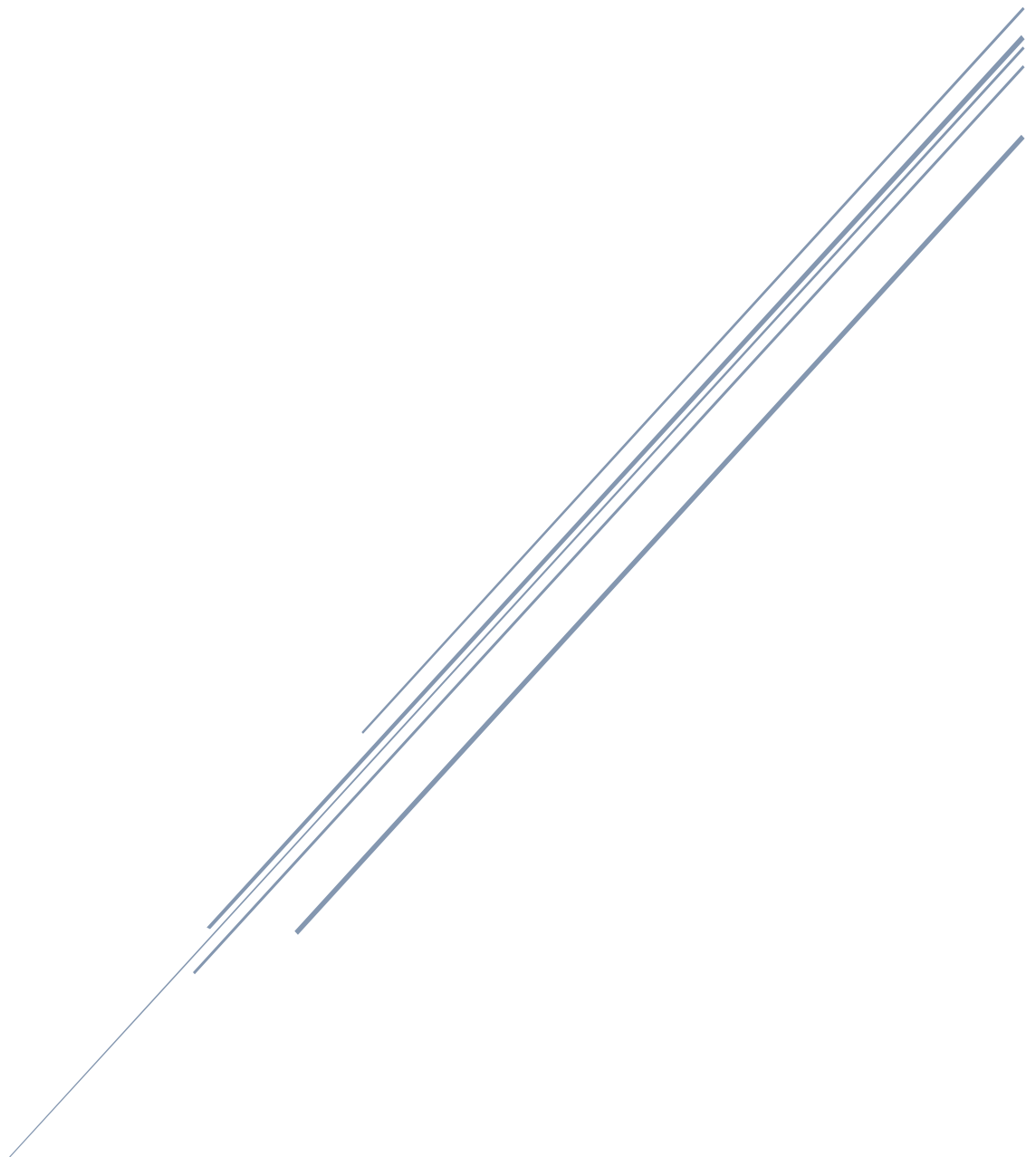


EISEN EN UITBREIDINGEN ICSS-B_COMPILER

Rick Jellema – 1614621 – maart 2022/2023

Docent: Niek van Diepen



Hogeschool van Arnhem en Nijmegen
Algoritmes, programmeertalen en paradigma's

Inhoud

Inleiding	2
1. Algemene eisen opgesteld door de HAN.....	3
2. Bugs	3
3. Extra uitbreidingen	4
4. Voorbeelden van product.....	5
4.1. Werkende voorbeeld van product.....	5
4.2. <i>Voorbeeld van uitbreiding 1</i>	5
4.3. <i>Voorbeeld van uitbreiding 2</i>	6
4.4. <i>Voorbeeld van uitbreiding 3</i>	6
4.5. <i>Voorbeeld van uitbreiding 4</i>	7
4.6. <i>Voorbeeld van uitbreiding 5</i>	7
4.7. Voorbeeld van uitbreiding 6	8
4.8. Voorbeeld van uitbreiding 7	8
4.9. Voorbeeld van uitbreiding 8	10
4.10. Voorbeeld van uitbreiding 9	11
4.11. Voorbeeld van uitbreiding 10	12
4.12. Voorbeeld van uitbreiding 11, 12, 13 en 14	13

Inleiding

Dit document bevat de gekozen uitbreidingen die geïmplementeerd zijn in de ICSS Compiler opdracht van het vak APP. Er wordt verwezen naar het “assignments.md” bestand waar in alle vereiste implementaties staan, dit bestand is aangeleverd door de HAN en beschikbaar in de bijgeleverde zip op de GitHub repository van Michel Koolwaaij

1. Algemene eisen opgesteld door de HAN

Het beroepsproduct heeft alle implementatie eisen van de types MUST & SHOULD uit het bestand “assignments.md” geïmplementeerd.

2. Bugs

Op het moment van schrijven is er 1 bug gevonden door de student.

- Bij calculaties loopt het programma vast als er booghaken gebruikt worden in de berekening (lijkt op een oneindige lus, wat leidt tot een `java.lang.OutOfMemoryError` exceptie), neem als voorbeeld de calculatie bij $500px + 20px + (50px * 2)$.

3. Extra uitbreidingen

Er zijn in het beroepsproduct enkele uitbreidingen toegevoegd. Deze uitbreidingen zijn goedgekeurd door de heer van Diepen. De student heeft zelf het aantal punten per uitbreiding aangegeven. Deze punten zijn door de student ingevuld op basis van de moeilijkheidsgraad en moeilijkheden tijdens implementatie.

#	Omschrijving	Punten
1	Regelnummer wordt aangegeven bij foutmeldingen gegeven door de checker.	2
2	Iedere variabele mag alleen een vast type hebben. Dan mag <code>Var := 10px;</code> en daarna <code>Var := 5%;</code> niet voorkomen.	3
3	Type checking op property values correcte waardes wordt toegepast. <code>Width: 100px;</code> mag, <code>Width:100;</code> geeft een error.	3
4	Onbekende property wordt aangegeven in de foutmelding.	1
5	Er kunnen enums toegevoegd worden aan <code>ASTNodes errors</code> om zo foutmeldingen met een <code>ERROR</code> of <code>WARN</code> of <code>SYNTAX_ERROR</code> aan te geven in de GUI wanneer de <code>Parsed AST</code> gechecked wordt.	2
6	If- en ElseClauses body wordt op de correcte plaats in de body van een selector gezet gezet bij transformatie.	4
7	Color- en boolean literal in een berekening geeft een foutmelding.	2
8	Meerdere selectoren in 1 lijn, zoals: <code>div, p, a { }</code> .	2
9	Comments maken is mogelijk.	0
10	<code>TRUE</code> , <code>True</code> , <code>true</code> , <code>FALSE</code> , <code>False</code> , <code>false</code> zijn geaccepteerde bool literal values. (puur omdat ik het hinderlijk vond om <code>TRUE/FALSE</code> in hoofdletters te schrijven, ik vind dat <code>TRUE == True == true FALSE == False == false</code>)	0
11	Mogelijkheid om tags te selecteren die 1 of meerdere classes gebruiken (<code>div.main.container</code> , selecteert alle divs met de css classes <code>main</code> en <code>container</code>)	1
12	Mogelijkheid om tags met identifiers te selecteren die 1 of meerdere classes gebruiken (<code>#identfier.main.container</code> , selecteert alle elementen met de identifier <code>#identfier</code> en de css classes <code>main</code> en <code>container</code>)	1
13	Mogelijkheid om alle elementen te selecteren die 1 of meerdere classes gebruiken (<code>.OuterContainer.InnerContainer</code> , selecteert alle elementen met de css classes <code>OuterContainer</code> en <code>InnerContainer</code>)	1
14	Het is nu mogelijk om css classes of identifiers te maken met hoofdletters in de naam en underscores (<code>main_container</code> , <code>Main_Container</code> , <code>MainContainer</code> , <code>MalnCoNtAlnEr</code> ← waarom je die laatste zou willen weet ik niet maar het kan in CSS en dus nu ook in ICSS)	1
Totaal aantal punten (max 20 punten te behalen)		20

Tabel 1: Uitbreidingen

4. Voorbeelden van product.

4.1. Werkende voorbeeld van product

The screenshot shows the ICSS Tool interface with the title bar "ICSS Tool September 2020, version 1". The interface is divided into three main panels: "Input (ICSS)", "Internal (AST)", and "Output (CSS)".

Input (ICSS):

```
width: ParWidth;
if[AdjustColor] {
  color: #124532;
  if[UseLinkColor]{
    background-color: LinkColor;
  } else {
    background-color: #000000;
  }
}
height: 20px;
}
a {
  color: LinkColor;
}

#menu {
  width: ParWidth + 20px;
}
```

Internal (AST):

- ▼ Stylesheet
 - ▼ Stylerule
 - TagSelector p
 - ▼ Declaration
 - Property: (background-color)
 - Color literal (#ffffff)
 - ▼ Declaration
 - Property: (width)
 - Pixel literal (500)
 - ▼ Declaration
 - Property: (color)
 - Color literal (#124532)
 - ▼ Declaration

Output (CSS):

```
p {
  background-color: #ffffff;
  width: 500px;
  color: #124532;
  background-color: #000000;
  height: 20px;
}

a {
  color: #ff0000;
}

#menu {
  width: 520px;
}

.menu {
  color: #000000;
}
```

Below the panels is a "Pipeline" section with buttons: "Parse", "Check", "Transform", and "Generate". The "Generate" button is highlighted. Below the buttons, the status "Generating output..." and "Generating succeeded" is displayed.

4.2. Voorbeeld van uitbreiding 1

The screenshot shows the ICSS Tool interface with the title bar "ICSS Tool September 2020, version 1". The interface is divided into three main panels: "Input (ICSS)", "Internal (AST)", and "Output (CSS)".

Input (ICSS):

```
p {
  background-image: #ffffff;
}

a {
  width: MyVarUndeclared;
}
```

Internal (AST):

- ▼ Stylesheet
 - ▼ Stylerule
 - TagSelector p
 - ▼ Declaration
 - Property: (background-image)
 - Color literal (#ffffff)
 - ▼ Stylerule
 - TagSelector a
 - ▼ Declaration
 - Property: (width)
 - VariableReference (MyVarUndeclared)

Output (CSS):

The output panel is empty.

Below the panels is a "Pipeline" section with buttons: "Parse", "Check", "Transform", and "Generate". The "Check" button is highlighted. Below the buttons, the status "Checking..." is displayed, followed by error and warning messages:

```
ERROR: The property "background-image" cannot be resolved. At line: 2
SYNTAX_ERROR: A width property-value can only hold pixel literal or percentage literal types. At line: 6
WARN: Variable usage invalid. Is the variable not declared or possible declared in an unavailable scope? At line: 6
```

4.3. Voorbeeld van uitbreiding 2

ICSS Tool September 2020, version 1

File	Input (ICSS):	Internal (AST):	Output (CSS):
	<pre>Var:= 10%; Var:= 1500px; p { background-image: #ffffff; }</pre>	<ul style="list-style-type: none">▼ Stylesheet<ul style="list-style-type: none">▼ VariableAssignment (Var)<ul style="list-style-type: none">VariableReference (Var)Percentage literal (10)▼ VariableAssignment (Var)<ul style="list-style-type: none">VariableReference (Var)Pixel literal (1500)▼ Stylerule<ul style="list-style-type: none">TagSelector p▼ Declaration<ul style="list-style-type: none">Property: (background-image)Color literal (#ffffff)	

Pipeline: Parse **Check** Transform Generate

Checking...
ERROR: Variable of type "PERCENTAGE" can't hold a value of type "PIXEL". At line: 2
ERROR: The property "background-image" cannot be resolved. At line: 5

4.4. Voorbeeld van uitbreiding 3

ICSS Tool September 2020, version 1

File	Input (ICSS):	Internal (AST):	Output (CSS):
	<pre>p { width: #ffffff; color: 100%; }</pre>	<ul style="list-style-type: none">▼ Stylesheet<ul style="list-style-type: none">▼ Stylerule<ul style="list-style-type: none">TagSelector p▼ Declaration<ul style="list-style-type: none">Property: (width)Color literal (#ffffff)▼ Declaration<ul style="list-style-type: none">Property: (color)Percentage literal (100)	

Pipeline: Parse **Check** Transform Generate

Checking...
SYNTAX_ERROR: A width property-value can only hold pixel literal or percentage literal types. At line: 2
SYNTAX_ERROR: A color property-value can only hold color literal type. At line: 3

4.5. Voorbeeld van uitbreiding 4

ICSS Tool September 2020, version 1

File

Input (ICSS):	Internal (AST):	Output (CSS):
<pre>p { onbekende-property-ofzo: #ffffff; font-color: #ffffff; }</pre>	<ul style="list-style-type: none">▼ Stylesheet<ul style="list-style-type: none">▼ Stylerule<ul style="list-style-type: none">TagSelector p<ul style="list-style-type: none">▼ Declaration<ul style="list-style-type: none">Property: (onbekende-propColor literal (#ffffff)▼ Declaration<ul style="list-style-type: none">Property: (font-color)Color literal (#ffffff)	

Pipeline: Parse Check Transform Generate

Checking...

ERROR: The property "onbekende-property-ofzo" cannot be resolved. At line: 2

ERROR: The property "font-color" cannot be resolved. At line: 3

4.6. Voorbeeld van uitbreiding 5

ICSS Tool September 2020, version 1

File

Input (ICSS):	Internal (AST):	Output (CSS):
<pre>p { onbekende-property-ofzo: #ffffff; font-color: #ffffff; width: UndeclaredVar; }</pre>	<ul style="list-style-type: none">▼ Stylesheet<ul style="list-style-type: none">▼ Stylerule<ul style="list-style-type: none">TagSelector p<ul style="list-style-type: none">▼ Declaration<ul style="list-style-type: none">Property: (onbekende-propColor literal (#ffffff)▼ Declaration<ul style="list-style-type: none">Property: (font-color)Color literal (#ffffff)▼ Declaration<ul style="list-style-type: none">Property: (width)VariableReference (Unde	

Pipeline: Parse Check Transform Generate

Checking...

ERROR: The property "onbekende-property-ofzo" cannot be resolved. At line: 2

ERROR: The property "font-color" cannot be resolved. At line: 3

SYNTAX_ERROR: A width property-value can only hold pixel literal or percentage literal types. At line: 4

WARN: Variable usage invalid. Is the variable not declared or possible declared in an unavailable scope? At line: 4

4.7. Voorbeeld van uitbreiding 6

ICSS Tool September 2020, version 1

File	Input (ICSS):	Internal (AST):	Output (CSS):
	<pre>UseColorWhite:=true; p { height:100px *2; if[UseColorWhite]{ color:#ffffff; } else { color:#000000; } width:100px; } p { height:100px; background-color: #ffffff; if[UseColorWhite]{ color:#ffffff; } else { color:#000000; } width:100px; }</pre>	<pre>▼ Stylesheet ▼ Stylerule TagSelector p ▼ Declaration Property: (height) Pixel literal (200) ▼ Declaration Property: (color) Color literal (#ffffff) ▼ Declaration Property: (width) Pixel literal (100) ▼ Stylerule TagSelector p ▼ Declaration Property: (height) Pixel literal (100) ▼ Declaration Property: (background-color)</pre>	<pre>p { height: 200px; color: #ffffff; width: 100px; } p { height: 100px; background-color: #ffffff; color: #ffffff; width: 100px; }</pre>

Pipeline:

Generating output...
Generating succeeded

4.8. Voorbeeld van uitbreiding 7

ICSS Tool September 2020, version 1

File	Input (ICSS):	Internal (AST):	Output (CSS):
	<pre>UseColorWhite:=true; ColorVar:=#ffffff; Calc:=ColorVar*UseColorWhite;</pre>	<pre>▼ Stylesheet ▼ VariableAssignment (UseColorWhite) VariableReference (UseColorWhite) Bool Literal (TRUE) ▼ VariableAssignment (ColorVar) VariableReference (ColorVar) Color literal (#ffffff) ▼ VariableAssignment (Calc) VariableReference (Calc) ▼ Multiplication VariableReference (ColorVar) VariableReference (UseColorWhite)</pre>	<pre>p { height: 200px; color: #ffffff; width: 100px; } p { height: 100px; background-color: #ffffff; color: #ffffff; width: 100px; }</pre>

Pipeline:

Checking...
ERROR: Variable assignment is invalid and thus not possible because of an invalid expression at line: 4
ERROR: Using color literals or boolean literals in Mutliplication, Addition or Subtraction operations is not allowed. At line: 4

4.9. Voorbeeld van uitbreiding 8

ICSS Tool September 2020, version 1

Input (ICSS):	Internal (AST):	Output (CSS):
<pre>AdjustColor := TRUE; A:= True; B:= true; C:= FALSE; D:= False; E:= false; p, .container, #main { color:#ffffff; background-color:#000000; } a, #anchor, .link { color:#aaaaaa; background-color:#101010; }</pre>	<pre>▼ Stylesheet ▼ Stylerule TagSelector p ClassSelector .container IdSelector #main ▼ Declaration Property: (color) Color literal (#ffffff) ▼ Declaration Property: (background-color) Color literal (#000000) ▼ Stylerule TagSelector a IdSelector #anchor ClassSelector .link ▼ Declaration Property: (color) Color literal (#aaaaaa) ▼ Declaration Property: (background-color) Color literal (#101010)</pre>	<pre>p, .container, #main { color: #ffffff; background-color: #000000; } a, #anchor, .link { color: #aaaaaa; background-color: #101010; }</pre>

Pipeline:

Generating output...
Generating succeeded

4.10. Voorbeeld van uitbreiding 9

ICSS Tool September 2020, version 1

Input (ICSS):	Internal (AST):	Output (CSS):
<pre>LinkColor := #ff0000; ParWidth := 500px; AdjustColor := True; UseLinkColor := true; // p html tag selector // used to give p tag some css flair... p { background-color: #ffffff; width: ParWidth; if[AdjustColor] { color: #124532; } /* UseLinkColor bool is used to quickly switch between #000000 or LinkColor... */ if[UseLinkColor]{ background-color: LinkColor; } else { background-color: #000000; } height: 20px; } a { color: LinkColor; } #menu { width: ParWidth + 20px; } .menu { color: #000000; background-color: LinkColor; }</pre>	<p>▼ Stylesheet</p> <ul style="list-style-type: none">► Stylerule► Stylerule► Stylerule► Stylerule	<pre>p { background-color: #ffffff; width: 500px; color: #124532; background-color: #ff0000; height: 20px; } a { color: #ff0000; } #menu { width: 520px; } .menu { color: #000000; background-color: #ff0000; }</pre>

Pipeline:

Generating output...
Generating succeeded

4.11. Voorbeeld van uitbreiding 10

ICSS Tool September 2020, version 1

Input (ICSS):	Internal (AST):	Output (CSS):
<pre>AdjustColor := TRUE; A:= True; B:= true; C:= FALSE; D:= False; E:= false; p { if[AdjustColor]{ color:#ffffff; } else { color:#000000; } if[A]{ color:#111111; } if[B]{ color:#222222; } if[C]{ color:#333333; } if[D]{ color:#444444; } if[E]{ color:#555555; } }</pre>	<pre>▼ Stylesheet ▼ Stylerule TagSelector p ▼ Declaration Property: (color) Color literal (#ffffff) ▼ Declaration Property: (color) Color literal (#111111) ▼ Declaration Property: (color) Color literal (#222222)</pre>	<pre>p { color: #ffffff; color: #111111; color: #222222; }</pre>

Pipeline: Parse Check Transform **Generate**

Generating output...
Generating succeeded

4.12. Voorbeeld van uitbreiding 11, 12, 13 en 14

ICSS Tool September 2020, version 1

Input (ICSS):	Internal (AST):	Output (CSS):
<pre>div.MainContainer { color:#ffffff; } body.main-container.Outer_Container { color:#ffffff; } #IdEnTiFiEr.class.hoverclass{ color:#ffffff; } .classContainer, a, .anchorClass{ color:#ffffff; } .maincontainer.oldcontainer.subjectcontainer.orsomething { color:#ffffff; }</pre>	<pre>▼ Stylesheet ▼ Stylerule TagSelector div.MainContainer ▼ Declaration Property: (color) Color literal (#ffffff) ▼ Stylerule TagSelector body.main-container.Outer_C ▼ Declaration Property: (color) Color literal (#ffffff) ▼ Stylerule IdSelector #IdEnTiFiEr.class.hoverclass ▼ Declaration Property: (color) Color literal (#ffffff) ▼ Stylerule ClassSelector .classContainer TagSelector a ClassSelector .anchorClass ▼ Declaration Property: (color) Color literal (#ffffff) ▼ Stylerule ClassSelector .maincontainer.oldcontainer ▼ Declaration Property: (color) Color literal (#ffffff)</pre>	<pre>div.MainContainer { color: #ffffff; } body.main-container.Outer_Container { color: #ffffff; } #IdEnTiFiEr.class.hoverclass { color: #ffffff; } .classContainer, a, .anchorClass { color: #ffffff; } .maincontainer.oldcontainer.subjectcontainer.orsomething { color: #ffffff; }</pre>

Pipeline:

Generating output..
Generating succeeded