The WyScript Language Specification

David J. Pearce
School of Engineering and Computer Science
Victoria University of Wellington, New Zealand
djp@ecs.vuw.ac.nz

December 4, 2013

Contents

Chapter 1

Introduction

1.1 Overview

1.2 Goals

1.3 History

Chapter 2

Syntax

$prgrm$	::=	(d)*	*// WyScript Program*
d	::=		*// Declarations*
	|	`type` `ID` `is` t `newline`	*// Type Declarations*
	|	`constant` `ID` `is` e `newline`	*// Constant Declarations*
	|	t `ID` `(` [t `ID` (`,` t `ID`)*] `)` `:` B `newline`	*// Function Declarations*
B	:==	(`indent` m)*	*// Blocks*
m	:==		*// Statements*
	|	`return` [e] `newline`	*// Return Statements*
	|	`print` e `newline`	*// Print Statements*
	|	`if` e `:` `newline` B [`else` `:` `newline` B]	*// If Statements*
	|	`while` e `:` `newline` B	*// While Statements*
	|	`for` `ID` `in` e `:` `newline` B	*// For Statements*
	|	`ID` `(` [e (`,` e)*] `)` `newline`	*// Invoke Statements*
	|	t n [`=` e] `newline`	*// Variable Declaration Statements*
	|	(n | e_1 `[` e_2 `]` | e `.` `ID`) `=` e `newline`	*// Assignment Statements*

Figure 2.1: Syntax for Declarations, Blocks and Statements

e	$::=$		*// Expressions*
	$\vert$	b	*// Boolean constants*
	$\vert$	i	*// Integer constants*
	$\vert$	r	*// Rational constants*
	$\vert$	c	*// Character constants*
	$\vert$	s	*// String constants*
	$\vert$	n	*// Variables*
	$\vert$	$\boxed{\texttt{null}}$	*// Null constant*
	$\vert$	$\boxed{\texttt{(}}\ e\ \boxed{\texttt{)}}$	*// Bracketed expressions*
	$\vert$	$\boxed{\texttt{\vert}}\ e\ \boxed{\texttt{\vert}}$	*// Size expressions*
	$\vert$	$e_1\ \boxed{\texttt{[}}\ e_2\ \boxed{\texttt{]}}$	*// Index Expressions*
	$\vert$	$e\ \boxed{\texttt{[}}\ e_1 \boxed{\texttt{..}} e_2\ \boxed{\texttt{]}}$	*// Range Expressions*
	$\vert$	$e\ \boxed{\texttt{.}}\ \boxed{\texttt{ID}}$	*// Access Expressions*
	$\vert$	$\boxed{\texttt{ID}}\ \boxed{\texttt{(}}\ [\ \overline{e}\ (\ \boxed{\texttt{,}}\ e\ )^*\ ]\ \boxed{\texttt{)}}$	*// Invocation expressions*
	$\vert$	$\boxed{\texttt{[}}\ [\ e_1\ (\ \boxed{\texttt{,}}\ e_i\ )^*\ ]\ \boxed{\texttt{]}}$	*// List expressions*
	$\vert$	$\boxed{\{}\ [\ \boxed{\texttt{ID}}\ \boxed{\texttt{:}}\ e_1\ (\ \boxed{\texttt{,}}\ \boxed{\texttt{ID}}_i\ \boxed{\texttt{:}}\ e_i\ )^*\ ]\ \boxed{\}}$	*// Record Assignment expressions*

Figure 2.2: Syntax for Expressions

	$\mid$	$n\ \boxed{\texttt{is}}\ t$	*// Instance of Expressions*
	$\mid$	$\boxed{\texttt{(}}\ t\ \boxed{\texttt{)}}\ n$	*// Cast Expressions*
	$\mid$	$(\ \boxed{\texttt{-}} \mid \boxed{\texttt{!}} \mid \boxed{\texttt{\&}} \mid \boxed{\star}\ )\ e$	*// Unary expressions*
	$\mid$	$e_1\ [\ op^2\ e_2\ ]^+$	*// Binary expressions*
op^2	$::=$	$\boxed{\texttt{+}} \mid \boxed{\texttt{-}} \mid \boxed{\star} \mid \boxed{\texttt{/}} \mid \boxed{\texttt{\%}} \mid \boxed{\texttt{++}}$	*// Binary Operators*
	$\mid$	$\boxed{\texttt{\&\&}} \mid \boxed{\texttt{\vert\vert}} \mid \boxed{\texttt{>}} \mid \boxed{\texttt{>=}} \mid \boxed{\texttt{<}} \mid \boxed{\texttt{<=}} \mid \boxed{\texttt{==}} \mid \boxed{\texttt{!=}}$	*// Binary Comparisons*
t	$::=$		*// Types*
	$\mid$	$\boxed{\texttt{ID}}$	*// User-Defined Types*
	$\mid$	$t\ [\ \boxed{\vert}\ t\ ]^+$	*// Union Types*
	$\mid$	$\boxed{\texttt{[}}\ t\ \boxed{\texttt{]}}$	*// List Types*
	$\mid$	$\boxed{\{}\ [\ t_1\ n_1\ (\ t_i\ n_i\ )^*\ ]\ \boxed{\}}$	*// Record Types*
	$\mid$	$\boxed{\texttt{void}} \mid \boxed{\texttt{null}} \mid \boxed{\texttt{bool}} \mid \boxed{\texttt{char}} \mid \boxed{\texttt{string}} \mid \boxed{\texttt{int}} \mid \boxed{\texttt{real}}$	*// Base Types*

Figure 2.3: Syntax for Operations and Types