



DUBLIN INSTITUTE  
*of* TECHNOLOGY  
*Institiúid Teicneolaíochta Bhaile Átha Cliath*

# IMPLMENTING A INTERPRETER FOR A SCRIPTING LANAGUGE USING HASKELL

## INTERIM REPORT

Zhen LAO

`zhen.lao@student.dit.ie`

*Supervisor:* Richard LAWLOR

*2nd Reader:* Cindy LIU

March 22, 2011

This Report is submitted in partial fulfillment of the requirements for the award of the degree of **BSc Computer Science** of the School of Computing, College of Sciences and Health, Dublin Institute of Technology.

## **Abstract**

In this thesis ,we focus on

**Keywords:**programming language YUN,Haskell,Parsec,plan,CFG,EBNF

## Declaration

I **Zhen Lao** hereby declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed \_\_\_\_\_  
*Zhen Lao*

## Acknowledgements

I would like to thank my supervisor Richard Lawor, for his valuable advice and useful suggestions on my project.

I am also deeply indebted to all the other tutors and teachers in Computer Science for their direct and indirect help to me.

Special thanks should go to my friends who have put considerable time and effort into their comments on the draft.

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Objective . . . . .	5
1.2	Introduction to Haskell . . . . .	5
1.3	Methodology . . . . .	5
<b>2</b>	<b>Grammar Design</b>	<b>6</b>
2.1	The Hierarchy of Grammars . . . . .	6
<b>3</b>	<b>Future Work and Project Plan</b>	<b>7</b>
3.1	Future Work . . . . .	7
3.2	Project Plan . . . . .	8

# List of Figures

# Chapter 1

## Introduction

### 1.1 Objective

The objective of this project is to develop an week-type interpreted language using Haskell.This language is able to support the following feature,

- basic for loop and while loop
- basic if-else statement
- functional invocation
- arbitrary dimension array
- polymorphic array

Furthermore, in project,the monadic design approach is applied as Haskell is different from other object oriented language.

### 1.2 Introduction to Haskell

Haskell is an advanced purely-functional programming language.By applying the used of Haskell to this project ,I have significantly reduce the coding time and spent most of my time to the design phrase.

### 1.3 Methodology

Agile development methodology is used in the entire development process.This project has been initially identified multiple iteration and each iteration contains three major stages research , development and testing.

## Chapter 2

# Grammar Design

### 2.1 The Hierarchy of Grammars

Noam Chomsky has describe three model of grammar [”Three models for the description of language”] and this grammar model has significantly effect the design of computer programming language.

Chomsky define a set of rule upon the formal grammar and categorize them into different levels.

A formal grammar of this type consists of:

- a finite set of terminal symbols.
- a finite set of non-terminal symbols.
- a finite set of project rules.
- a start symbol.

From previous formal grammar definition, legitimate production rules can be written as

$$S \mapsto aS \text{ and } S \mapsto ab$$

In this example,we can assume that the grammar consists of two projection rules and the starting symbol is  $S$ .The terminal symbols are lower letters  $\{a, b\}$  . From this example, If we start from the either rule 1 or rule 2 ,we could derive a grammar of  $\{a^n b | n > 1\}$  ,which can be enumerate like  $\{aab, aaab, aaaab, \dots\}$  .

The Chomsky hierarchy consists of the 4 levels:

- Type-0 grammars. It is a unrestricted grammars that include all.
- Type-1 grammars.
- Type-2 grammars.
- Type-3 grammars.



## Chapter 3

# Future Work and Project Plan

### 3.1 Future Work

I have done most of the research work of the project. the future work will be implementing the actual parser. There will be an initial implementation of part of the EBNF definition. The execution engine will be implemented at the same as the parser. Due to Haskell have a powerful abstraction mechanism and a suitable library (Parsec), the implementation will not be too difficult.

The major barrier will be implementing the execution engine since I have no knowledge about it. There will be more research in the later stage of this project on the interpreter part as well as the grammar and Haskell part.

## 3.2 Project Plan

<b>Before November</b>
I have finish most of research on Chomsky's CFG grammar and Haskell.I have started to implement a prototype of the interpreter.
<b>November 12 to November 26</b>
Research on the execution engine of the interpreter.
<b>November 12 to November 31</b>
Development a prototype by following the documentation of parsec. The document "Write Yourself a Scheme in 48 Hours/Parsing" offer an example to implements a interpreter for Scheme using parsec library of Haskell.
<b>November 31 to January 15</b>
Implement a subset of EBNF specification of <i>yun</i> .Add the error checking to the interpreter.Meantime,as the code growing ,unit test will be added to guarantee the quality of existing code.
<b>January 16 to February 15</b>
Implement all the EBNF specification of <i>yun</i>
<b>February 31 to March 15</b>
Implement the IO command (library).Add more test code.
<b>February 16 to March 16</b>
Review the EBNF of <i>yun</i> programming language.Implement more library for it.Start the system testing.
<b>March 17 to April 8</b>
Prepare for the project fair.