

Language Engineering

Introduction :: TANotes

1 Introduction

Welcome to COMS22201: Language Engineering! This course is about programming languages, how we can implement and extend them, how computers understand them as well as special language features.

This set of notes outlines the course structure during Teaching Block 1 and give some overview about assessments, resources and where to go for help. This part will be taught by Dr. Nicolas Wu, also known as Nick.

2 Course Content

The following topics will be covered in this course:

1. Domain-specific languages: languages specialised for very specific purpose, but not as powerful as general purpose languages, e.g. HTML.
2. Parser combinators: functions that take input strings and convert them to an abstract syntax tree that can then be analysed and executed.
3. Effect handlers & the *Free* monad: functions that take a syntax tree and handle it, i.e. compiler, turning syntax into instructions.

3 Assessment

This part of the unit is 100% exam; however, there are other components that should be completed as well, to both learn new content and assess your current understanding.

The coursework will involve writing a parser and compiler for TinyBASIC. The coursework will be released right at the start and you are free to spend as much time as you wish on it, as it is purely a learning exercise. The coursework will be entirely formative (unassessed), but is essential and will be checked. Failure to show proof of attempt may result in failing the course...

In addition to the coursework, there will be weekly worksheets to complement the lectures. These are to be completed during the lab sessions, and must be signed off by TAs. They will assess content covered during that week, and should be manageable with help (if needed). Some questions are specifically aimed to stretch you, so don't worry about getting every question right.

4 Resources

There will be 2 lectures per week, for a total of 22 lectures. In addition, there will be a 2 hour lab each week, where you will be expected to complete the lab worksheets and ask for any help with the coursework or any other part of the course. This is the best method to ask for assistance or clarification if needed, both from the professor and the TAs. In addition, there will be office hours scheduled, where you can go and clarify some problem or ask for help.

In addition to all of these, there are TA notes on each of the topics that give an understanding of each topic from a different approach than the lectures. These are excellent sources to further your understanding about a topic, or fill gaps in your knowledge.

5 Information on Worsheets and Notes

The difficulty of questions will be indicted using a dagger system:

No dagger	These questions should be fairly straight forward.
†	These questions should require some more thought.
††	These questions will be very challenging.
†††	The TAs may not be able to do these.

These sheets are rendered using more mathematical notation. Here is a guide of how to type certain symbols in Haskell:

λ	<code>\</code>	backslash
\circ	<code>.</code>	full stop
\neg	<code>not</code>	not
\rightarrow	<code>-></code>	dash, greater than
\Rightarrow	<code>=></code>	equals, greater than
\vee	<code> </code>	pipe, pipe
\wedge	<code>&&</code>	ampersand, ampersand
\equiv	<code>==</code>	equals, equals
in°	<code>inOp</code>	inOp
$\langle * \rangle$	<code><*></code>	less than, asterisks, greater than
$\langle * \rangle$	<code><*</code>	less than, asterisks
$\rangle *$	<code>*></code>	asterisks, greater than
$\langle \$ \rangle$	<code><\$></code>	less than, dollar, greater than
$\langle \$ \rangle$	<code><\$</code>	less than, dollar
$:+:$	<code>:+:</code>	colon, plus, colon