

# COMP712 Programming Languages

## Interpreter Implementation Project

### Description

The aim of this project is to implement an interpreter for a programming language called AJS (A JavaScript) which is a subset of ECMAScript 2018. The syntax and related descriptions of this language is provided in a separate document.

The interpreter is to be implemented in Racket with the help of the **SLLgen** scanner and parser generator and the **Datatype** macro extension introduced by D. Friedman and M Wand in their book *Essentials of Programming Languages*, third edition.

A file called **AJS\_test\_cases.ajs** is provided with some AJS source code for testing your interpreter. The test cases are divided into 4 parts:

1. Simple arithmetic and constant declarations
2. Function declarations and compound functions
3. Booleans and conditionals
4. Recursive functions

In your development, you may choose to first implement the simpler aspects of the language first. For example, you may implement that part of AJS that can only pass the tests in Part 1. Make sure that it works before proceeding to include more syntax to enable your interpreter to pass the tests in Part 2, etc. Marks are allocated to the language features in each of these four parts.

Note that your interpreter does not have to provide the ability to read program code from a file (although it would be nice). But it has to provide a REPL where the users can enter their code.

This project accounts for 50% of the total marks for this course.

### Submission

Students can work on their own or form groups of two. In both cases, you will need to self sign-up to a group on Canvas.

Each group is required to submit the following:

1. Documentation of your implementation including:
  - Verification that the grammar is LL(1)
  - If you are not able to implement the full AJS language as described, then clearly describe which part of it is implemented and provide the syntax of the implemented language
  - Description of additional data structures that you have used to support your implementation
  - Describe the approach and design choices you have made in your implementation

- Description of how your interpreter is to be run (so that I know how to use it for my tests)
2. Program source code – should contain adequate comments.

**The deadline for submission is specified on Canvas.** If your group requires extension, please contact me at least two days before the deadline. Note that due to the tight schedule by which the marks have to be submitted to the School and Faculty Exam Board, very limited extension is possible.

#### Mark Allocation

|                          |             |
|--------------------------|-------------|
| Documentation            | 20%         |
| Part 1 language features | 20%         |
| Part 2 language features | 20%         |
| Part 3 language features | 20%         |
| Part 4 language features | 20%         |
| <b>Total</b>             | <b>100%</b> |

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs