Logo, company name

Description automatically generatedCSCI415

Compiler Design

**Compiler Project phase 1**

Build a scanner for the **JASON programming language:**

* + Language is case-insensitive
  + All Keywords are reserved (Shown in upper case for emphasis)
  + Parenthesis re not used in a CALL statement if no parameters are being passed.
  + JASON has 4 arithmetic operators (+,-,\* and /) and 4 relational operators (=,>,<,!). The (!) is used to denote “is not equal to”
  + The entire operator set consists of one-character operators
  + Comments are enclosed in braces {}

**JASON has valid types of statements:**

* READ Identifier
* WRITE Identifier
* Begin Statement
* End statement
* Declare statement
* Data type identifier
* SET Identifier = Expression
* IF Condition THEN Statement(s) ELSE Statement(s) ENDIF
* WHILE Condition DO Statement(s) ENDWHILE
* UNTIL Condition DO Statement(s) ENDUNTIL
* CALL Identifier(ArgList)

**Token classes:**

* You should identify the tokens classes in a way that will help you in the following phase (parser)
* You can use any programming language you prefer for building the scanner.
* You can use flex or any other similar tool where you declare the regex rules and the tool takes input code and outputs the tokens.
* Phase 1 discussion : 9/12 (and it will not be extended).

**Example scanner input and output(the output will depend on the token classes you will identify)**

**Input code:**

Integer x;

Set x = 1;

|  |
| --- |
| Tokens |
| Datatype |
| Id |
| Semicolon |
| Set |
| Id |
| Equal |
| Num\_constant |
| Semicolon |