Project 1

Due: Tuesday, 23 November, 2021 by 11:59 PM

A. Overview.

In this project, you (and hopefully a partner) will design a language of your own, describe its grammar, implement a translator for it, and provide some sample programs. You will also look at another student language and write a couple programs in that language.

B. General Guidelines.

- All course policies should be followed (specifically see guidelines about late work and academic dishonesty.)
- You may work individually or with a partner on this project. You will submit only one project even if you are working with a partner. **Please work with a partner unless you have a very, very good reason not to.** You must let me know who your partner is, so I can make groups on D2L. **Please let me know who your partner is by Tuesday 19 October, 2021 at 11:59 PM.**
- The language you develop must be different from any other existing languages, but you can borrow ideas from languages you have seen.
- There is an opportunity for a big chunk of extra credit points in this assignment!

C. Specific Guidelines.

Part 1. The Grammar

Design a language and describe it with a grammar (like what we did in the first few weeks of class.) Your language must include the features below, and it must be **unambiguous**.

- integers
- variables
- variable assignment
- basic integer expressions with basic operators: addition, subtraction, multiplication, division, modulus
- booleans
- boolean expressions with basic operators: and, not, or
- comparison operators: greater than, less than, equal to (and others if you want)
- conditionals
- loops
- printing to output
- command line arguments
- string literals

Part 2. The Translator

Using your grammar, you must implement a translator that parses your language and produces a compilable/runnable program in another language. You may choose that language, but I

recommend using something you are very familiar with and something that has good regular expressions and pattern matching. (I used Java.)

The translator program must be able to take (as a command line argument) a file name for a file that contains a program written in your language. It should read that program in, parse it, and produce an output file that is a working program in the language of your choice. (We should then be able to compile/run that resulting program and get the desired results.)

Part 3. The Programs
You are required to submit 5 programs written in your language. These should be stored as text files (with the extension .txt) and make sure you name them according to the guidelines below.

- *Program1.txt*: This program should take in two command line arguments that are integers *x*, *y,* and *m*. It should then calculate and print out *the number of multiples of x* and *y* in the range 1 to *m*.

  Example: *Program1.txt 2 5 20* should print out “12”

- *Program2.txt:* This program should take in one command line argument that is an integer *x* and print out “prime” if *x* is prime and “not prime” if it is not prime.

  Example: *Program2.txt 7* should print out “prime”
  Example: *Program2.txt 8* should print out “not prime”

- *Program3.txt, Program4.txt,* and *Program5.txt*: Each of these programs should produce a parsing error. They are to show that your translator does not accept programs that are not syntactically correct (according to your grammar). Each one should illustrate a different parsing error.

Part 4. The Presentation
In this part, you will make an uploadable presentation that includes the following:
- Explain some of your syntactic choices when you were designing your language.
- Describe some of the challenges you faced when writing your translator and how you solved them.
- Give a tutorial about your language.
- Describe language design choices you would make if you were going to expand your language even further. (I recommend referencing other languages you have experienced here.)
- The presentation can be in any uploadable format, but let’s try to stick to common file types that work across different systems (e.g. PDF, .txt).

Part 5. More Programs (The deadline for this part is 30 November, 2021 at 11:59 PM).
In this portion, you will each be assigned to another student language. You will go through the presentation and write two small programs in that language. The content of these programs will

be announced after the submission deadline, but rest assured, they will only require the aspects that should be in your grammar.

D. Grading

Item	Points	General Criteria	Who is graded
Grammar	50	Complete, clear, unambiguous.	You and your partner
Translator & Initial Programs	75	● Translator compiles ● Translator complete ● Translator parses according to grammar ● Translator translates programs (or produces errors for bad syntax) ● Translated programs compile ● Translated programs produce good results	You and your partner
Presentation	50	Complete, clear, creative, shows that you thought through the language design.	You and your partner
Additional Programs	25	● Correct according to language syntax ● Translator translates & new program compiles and runs properly	You and the group who created the language

E. Extra Credit

What is described above are the requirements for the project. However, you have the opportunity to earn *up to 50 extra credit points* by adding additional features to your language. This could be adding more types, adding a type system, adding aggregate items, adding new parts to the grammar, adding more “compile-time” error handling, etc.

I strongly recommend that you focus on getting the overall requirements done first. After that, you can think about adding to the language. If you decide to try that, **you must meet with me first to discuss what you would like to add.**

A few other things:

- The points awarded for this will vary depending on what you actually add to the language.

- Anything you add should be reflected in the whole project. That is, you may have to add it to your grammar in Part 1, you will definitely have to provide some additional programs that illustrate the new feature, and you will definitely have to include it in your presentation.

F. Submission

Put all of your files in a zip drive and upload it to both D2L and Piazza. The Piazza upload can be anonymous. The deadline for both is Tuesday, 23 November, 2021 by 11:59 PM and every day late loses you 20 points, so I strongly recommend you don't turn it in late.

Submission for Part 5 can be done through Piazza and the deadline will be Tuesday, 30 November, 2021 by 11:59 PM. Please note that the grade for Part 5 will be for both the language designers and the programmer, so don't let your classmates down!

As always, if you have any questions or doubts, just ask! And have fun!