Algonquin College Logo

# SCHOOL OF ADVANCED TECHNOLOGY

### ICT - Applications & Programming

### Computer Engineering Technology – Computing Science



A11

Language Specification

Team:

[Sewmini Jayasinghe] - Id: [041057012] / [Rashin Babaee] - Id: [041043836]

Language Name [RS]

***This template is suggested (not mandatory) to answer A11 Specification.***

|  |  |
| --- | --- |
| **Part**  **1** | **Language User Reference** |

**EXPLANATION**

*The purpose of this assignment is to invent a new computer language.*

* *This language can have the syntax and structure of your choosing.*
* *Option 1: Adapt the ‘Julius’ language to be Julia compatible (see* [*https://julialang.org/*](https://julialang.org/)*).*
* *Option 2: Define a* ***DSL*** *– Proper to solve specific problems (ex: science, economy, music, etc.)..*
* *This is going to be a fairly basic language. There's a lot of functionality that we'll be skipping over, while we implement the basics. You will need to tell me those basics, of course. In this document, I'm going to explain the steps of what to do with a bit of detail.*
  1. **User Manual**

**Element 1: Name / Extension**

*[Name your language! We suggest you use one "word" related to your “Julia-like” language or DSL]*

*RS*

*[What is the filename extension of your language? For example, for C it is .c, and for Professor Paulo's “Julius”* ***language*** *it is ".****jul****".]*

*.rs*

*[What is your language patterned after, or what is it similar to? What languages are inspiring your choice? It's okay if you're following Julia closely.]*

*This language is Julia-like language.*

**Element 2 – Comments**

*[Comments: I want to do comments in your language. How do I write them?]*

If you want to comment in this language, it starts with # for single line comments and for multiline comments it starts with #= and ends with =#.

**Element 3 – Keywords**

*[Keywords: List the sequence of reserved / key words from your language]*

If, elseif, else, while, for, do while, true, false, break, return.

**Element 4 – Datatypes**

*[Datatypes: Define integers, real numbers (float points) and strings. Determine their ranges. ]*

*[Remember to define the number of bytes – and, if possible, range]*

 *For Integers, there will be 2-bytes assigned and the range will be -32000 to 32000*

 *For Floating-point numbers, there will be 4-bytes assigned and the range will be up to 5*

*decimal places.*

 *Later on, if more bytes needed, we can use long with them and it will add extra 2-bytes to the*

*size.*

*For strings, set of characters or a single character that is enclosed within brackets. It is a finite sequence of characters.*

**Element 5 – Variables**

*[Variables: How would a programmer define variables that can hold integer numbers (numbers with no decimal point), floating point numbers (numbers with a decimal point) or text (ie: strings in Java). This is element 1. Consider if you want to flag the variables in a special way, like SOFIA or BASIC, or not, like C or Java.]*

 *x = 10*

 *x = 10.987*

 x = “Hello World!”

**Element 6 - Commands**

* ***Attribution / assignment****: How does your language let a programmer assign a value to a variable? (Will you allow casting? If so, how will it work?) How will your language handle math, and will it allow strings to be concatenated (merged)?*
* *If it is a String, it should be*

*Ex:*

*a = “Hello World!”*

* *If it is a integer, it should be*

*Ex:*

*a = 200*

*Math:*

 *For math values, Variables can store integers with any sign like -10, +10. But the signs*

*should be used as prefix of value not suffix. 10- is illegal value. Any other sign except +/- will be*

*considered illegal while storing value in the variable.*

 *Especially for math calculations, there will be 5 signs used:*

 *+ will be sign for addition.*

 *- will be the sign for subtraction.*

 *\* will be the sign for multiplication.*

 */ will be the sign for division.*

 *% will be the modulus operator.*

*Concatenating String:*

 *In case of we concatenate String, we can use the \* operator. \* operator will be*

*placed between 2 strings and the result will be stored in another string variable.*

* ***Selection****: How does your language do if-style logic? (Optional: Do you want to do some kind of switch/case as well?). You will need to explain how "conditionals" work in your language. How do you write Boolean operations, such as "or", "and", "not", and other conditions, such as less than, greater than, etc?*

If name starts with A, it can print your name begins with A, else it shows your name does not begin with A.

name = readline()

if startswith(name, “A”)

println(“your name begins with A.”)

else

println(“your name does not begin with A.”)

end

*For conditional Expressions, we have 3 types*

 *AND – this will be the Logical AND operator.*

 *OR – this will be the Logical OR operator.*

 *NOT this will be the Logical NOT operator.*

*For Comparison Operators, the signs will be as follows:*

* *< For less than*
* *> for more than*
* *= for equal*
* ***Interaction****: How will your code handle looping? (You can do one or more of a for-style loop, a while/do loop, etc.)*

*we have the option to use FOR loop or WHILE loop or DO-WHILE loop.*

* ***Input****: How does your program get input from the keyboard? (Strings are easiest.)*

For getting input from keyboard, it is readLine()

* ***Output****: What would a programmer type to put output on the screen? What sort of variables or data will your code take?*

*For printing something on console, the command will be*

println(“Hello World”)

* ***Functions****: [Function definition: parameters and returning types]*
  + *What will be the syntax for making a function or subroutine?*
  + *How will it take parameters?*
  + *How will it return results?*
* For writing a function, () includes every parameter with comma to separate them.
* Function main with return 0 or 1.

Ex:

Function main ()

//statement

Return 0

end

**Element 7 – Proper elements**

*[Include specific features / elements to be included in your language]*

* *What you could include / modify? Think about new datatypes / structures / commands, etc.*

*We could imagine, defining a datatype before the variable. Because, for example, it could help the user or the programmer not to mix up a integer and a float number.*

* *Note: Do not share this info (it is supposed to be your proper elements in the language.*

|  |  |
| --- | --- |
| **Part**  **2** | **Examples** |

**Option 1: Julia-like**

**Hello World**

|  |  |  |
| --- | --- | --- |
|  | **# Ex 1: This program prints Hello, world!**  **print("Hello, world!")** |  |

**Sphere Volume Expression (or any other example)**

|  |  |  |
| --- | --- | --- |
|  | **# Ex 2: This program calculates the volume of a sphere**  **function sphere\_vol(r)**  # julia allows **Unicode names (in UTF-8 encoding)**  # so either "pi" or the symbol π can be used  return 4/3\*pi\*r^3  **end**  **vol = sphere\_vol(3)**  **# @printf allows number formatting but does not automatically # append the \n to statements, see below using Printf**  **@printf "volume = %0.3f\n" vol** |  |

*[TIP: See examples in the Lecture Notes –* ***Appendix 1****]*

**Option 2: DSL**

**[Your example here]**

|  |  |  |
| --- | --- | --- |
|  | [Your Code here] |  |

|  |  |
| --- | --- |
| **Part**  **3** | **Architectural Aspects** |

**Advantages**

*[What's the goal of your language? Are you trying to make something simple, fun, complicated? My personal language, Chambly, is based around being useful to scientists. (You can just make something up here, honestly. Think about it a little bit, have a little fun.)]*

*We made something simple here that is Julia-like. But more importantly our language will implement some good functionalities from other languages like java, C.*

**Strategy: C Implementation**

*[How your language can be implemented in C – ex: datatypes]*

* *In plain English, or maybe even some high-level pseudocode, how are you going to parse your language? You will be writing a compiler for your language, so these are some things you need to think about.*

*Because RS language is Julia-like, it is the same of Julia. So, it’s basically same as Julia implementing in C but with some different ideas such as defining a datatype before the variable.*

*[Your ideas about how to identify elements from language]*

* *Consider your "write to the console" command as an example. How will your compiler detect it? How will it sort out what to write to the console? What if there's some literal text (ie: "this is going to get printed") instead of variables?*

*We are thinking of using a temporary variable to store the text after reading the file then append to a string.*

*[Your ideas about how to identify scope (ex: blocks between conditionals or functions)]*

* *How do you mark a block of code? If I use your loop logic, how do I control what portion of code gets looped through? In C, you might use { and }. In Python, the indentation is what matters. How does it work in your language?*

*Because our language is similar to Julia, we use bracket { to start the loop and again bracket } to finish the loop. Every code between in these two brackets will be executed.*

**Basic ideas about C implementation**

*[Which structures or datatypes you imagine to use in your language implementation]*

* *What do you think is going to be really hard about this? What would be, in your opinion, the hardest part of parsing your own new language? You don't have to write an essay, a paragraph or two will be fine.*

*We think the hardest part will be switch and case in Julia, because there is not ant switch and case in Julia. So, we are trying to modify this language to have a better language.*

***Note 1: C Datatypes***

*Remember that you are implementing your language in ANSI C. For this reason, you cannot create arbitrarily your language (from scratch). You need to use what is already provided by C Compiler. For this reason, think about using and defining the language obeying the datatypes.*

**Problems when using C implementation**

*[Your vision about main problems / difficulties when implementing a new language (ex: memory allocation, range of datatypes]*

* *Range of datatype will be a problem because ranges in Julia and C are different.*
* *There will be a issue of memory allocation for example in arrays.*

**FINAL SUGGESTIONS**

*Here some ideas to think about your language....*

* *Don't make this assignment harder than it needs to be on yourself. Focus on making the syntax for your language that meets our requirements. Worry about extra features later.*
* *Don’t worry if your new language winds up having really difficult parts. You'll be allowed to change your language as you go along, as long as you make "patch notes" to explain those changes. We'll tell you about this later.*
* *There's a marking key at the end of* ***CST8152\_Compilers\_W23-A11-Specification*** *that should steer you along for grades. Focus your efforts on where you'll get the best results.*
* *Finally, think about creating an “master-piece”: until now, you have used several languages. And if you have conditions to define yours, how it could be?*

**References**

*[1]*  Prof. Svillen Ranev, ICT - Applications & Programming Computer Engineering Technology – Computing Science, Compilers lecture notes, Algonquin College, 2020.

[2] if...else statement, https://riptutorial.com/julia-lang/example/15208/if---else-statement.

* *NOTE: Even if you use any tool (ex: ChatGPT), report here.*

Algonquin College

Winter, 2023