Algonquin College Logo

# SCHOOL OF ADVANCED TECHNOLOGY

### ICT - Applications & Programming

### Computer Engineering Technology – Computing Science



A11

Language Specification

Team:

[Haoyun Deng] - Id: [041012253]

Language Name [cloud]

***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/>).*
* *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]*

*[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****".]*

*[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.]*

The suffix of the program language is named . cld

As a julia-like language, my language model will be very close to julia

**Element 2 – Comments**

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

I want to use “#comment” as a single line comment

I want to use “#=comment=#” for multi-line comments

**Element 3 – Keywords**

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

Scheduled keyword list:

begin, break, catch, const, continue, else, elseif, end, false, finally, for, function, global, if, local,while, return, true, try

“ ; ” is used to delimit the end of the previous statement

**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]*

There is a number type in the plan:

Int -2,147,483,648 —— 2,147,483,647 , 8byte

Char -128——127 , 1byte

Float 2.3E-308 —— 1.7E+308 , 8byte

String

Bool 0——1 , 1byte

Null, 4byte

Unsigned data types are allowed

Use the keyword Unsign before the data type

Example： Unsign Int

**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.]*

Variables don't need to be marked in a special way The syntax for defining a variable is: data type + variable name. Variables can be assigned a value at initialization time.Variable names can be uppercase and lowercase letters with symbols and numbers, but the letter must come first. Variable names do not allow spaces.

For example: Int variable1;

The definition of a constant is similar to a variable, but you need to add a keyword “const” in front of the data type

For example: const Int constants1=1;

**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)?*

It is planned to use "=" to assign values to variables. Casting is allowed, and characters and numbers can be converted through ASCII. Two strings can be combined using the + sign.

Casting example: x=’a’; y=Int(x);

Use the following operators to handle math:

+,-,\*,/,%,^,\,÷

* ***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?*

The conditional statement in the plan uses the keywords if else and elseif, and for the switch structure, use the switch and case keywords

Use the following boolean operators:

&&,!,||,<,>,!=,<=,>=,==

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

Plan to use 2 loops, for-do, while:

For-do: for expression end;

While： while expression end;

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

Use the readline keyword to enter, for example： readline(variablename)

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

Use the print keyword to output,If you need a newline, use println

for example： print(variablename/value/string)

* ***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?*

Use the following syntax to initialize a function:

Function(keyword) Functionname(parametername,parametername(optional))

::return data type(optional) end;

Example: Function fun1(var1,var2) ::Int

end;

The function name is Main is the main method

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

If you want to use a do-while loop, i planed a new keyword about while loop, the keyword is “do” if you add this word before your while loop, this loop will be executed once time.

Example: do while x<1

X=x+1;

End;

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

**Option 1: Julia-like**

**Hello World**

|  |  |  |
| --- | --- | --- |
|  | [Your Code here]  function Main()  print(‘Hello World!’);  end; |  |

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

|  |  |  |
| --- | --- | --- |
|  | [Your Code here]  function Main()  Float Pi=3.14,r,V;  readline(r);  V=4.0/3.0\*Pi\*r^3;  print(V);  end; |  |

*[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.)]*

My main goal is to make my programming language easier to use. Also, I would like to add some structures and process control like other GPL languages.

The main advantage of my language is that it can make the programmer familiar with "traditional programming languages" such as C or Java more comfortable with Julia.

**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.*

I would like to build some functionalities by C to implement some keywords that Julia has but C not. It will make my language get close with Julia.  
For example:  
In my language, ^ represents the Exponentiation. If there is a code: 3 ^ 5 was detected. Then the complier would invoke the following function:  
  
Pass in: The number before the ^, the number after the ^  
For I = 1 to The number after the ^  
The result = the number before the ^ x I  
End For  
Pass out: The final result

*[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?*

As you can see in the example above, the write will be the keyword to write the things to the console and switch the line. Once the keyword write is detected, the compiler will invoke the functions for console outputting. All the things that need to be written to the console will be put into the bracket. My solution to distinguish the variable and literal text is if the output target is literal, the developer should put them into apostrophes (‘’). By default, the program will treat all the arguments without apostrophes as a variable.

*[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?*

I would like to block the loop area by keyword “begin” and ”end”.

In Julia the loop block was implemented by begin and end keyword. The only modification in my language is it needs to add a semicolon after end.

For example:  
  
for I to 100  
print line “Hello World”  
end

**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.*

It's hard to implement some datatype that the C doesn't have, for example, a boolean value and String. I have to think about how to implement those myself such as memory allocation and how to manipulate the new type of data through expressions

***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]*

if I implement the assignment operation by prototyping functions in C, to make sure to allocate reasonable memory space for a variable, I have to use malloc manually to allocate the memory space for the variable. However, it will dramatically increase the usage of Heap. As a result, the total efficiency of my language will decrease.

**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**

*[Include eventual references used here]*

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

Algonquin College

Winter, 2023