Algonquin College Logo

# SCHOOL OF ADVANCED TECHNOLOGY

### ICT - Applications & Programming

### Computer Engineering Technology – Computing Science



A11

Language Specification

Team:

[Adam Di Cioccio] - Id: [041019241]

**Language Name [City / Country / Place]**

***This template is suggested 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. It's fine if you pattern it to follow the SOFIA example closely, but you may imitate other languages, or invent your own, as long as you're clear about the origins of what you're doing.*
* *Your choices are not set in stone. You may make changes to your language as we go along, as long as you make "patch notes" explaining what those changes are.*
* *Bear in mind that the purpose of this course is to create a compiler that will compile your language. So, I suggest not making things more complicated than they need to be.*
* *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. **General Language Overview**

**Name**

*ABRUZZO*

**Extension**

*.abr*

**Genealogy**

*Python and Java.*

**Advantages**

*The main advantage to my programming language is it will be very easy to read and understand for people learning how to program. I will remove unnecessary things similar to python that cause clutter and make the program harder to understand.*

**Hello World Example**

**out “Hello World!”**

* 1. **User Manual**

**Element 0 – Comments / Keywords**

**# This is a single line comment**

**## This**

**# is a**

**# multiple line**

**## comment.**

**Keywords: accepts, returns**

**Element 1 – Variables and Datatypes**

**int integerName = 10**

**dbl doubleName = 1.5**

**str stringName = “Hello”**

**Bytes needed are increased / decreased based on the size of given variables.**

**The range can vary anywhere from 8 bytes all the way up to 256 bytes.**

**Unlike python you must declare what the data is instead of the program assuming the type for flagging variables in a special way.**

**Element 2 - Commands**

**You will be able to assign values to a variables and even allow to parse through different variables. Math will be included in this language as well as string concatenations. (Math.root, Math.power, concat(string + string). )**

**The looping will be a mix of python / java as in the syntax using most of the same simple conventions but my language will make it even easier to use and understand**

**if (CONDITION) {}**

**elseif (CONDITION) {}**

**else {CONDITION}**

**for x -> (RANGE) {}**

**while (CONDITION) {}**

**in inputVariable;**

**out inputVariable**

**Element 3 - Modules**

**Similar to python functions are declared and can return any data type as you please.**

**You can either declare a function as void, int, dbl, str, bool and they must return the data type accordingly.**

**void function myFunction;**

**bool function myFunction return false;**

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

**Hello World**

|  |  |  |
| --- | --- | --- |
|  | void main()  out “Hello World!”  ; |  |

**Sphere Volume Expression**

|  |  |  |
| --- | --- | --- |
|  | void main()  dbl pi = 3.14  dbl input, output  in (input)  output = 4.0/3.0 \* pi \* (input \* input \* input)  out “Volume ” + output  ; |  |

*[TIP: See examples in the Lecture Notes – Appendix 4]*

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

**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.*

*You will be able to parse data types in my language as long as its done properly. Such as if you try to parse an integer to a string that will work but if you attempt to do the opposite with letter characters it will return error.*

*int newStr = parse(10) GOOD*

*int newInt = parse(“Hello”) BAD*

*[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?*

*All variables or literal text will be printed out using the ‘out’ function.*

*out variableName*

*out “Hello World!”*

*out “Name: ” + variableName*

*[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?*

*For any type of loop it will use the standard {} and for a function it will start at the line of the call and end at a semi-colon. In my program there will be no semi-colons after each statement meaning that indentation does matter and you must start a new line for each statement.*

**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.*

***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]*

*The biggest challenge I will face with this is understanding how to do this from scratch as I have never done or even thought about creating my own language before and this is all completely new to me. I do understand a handful of languages now and I think I can use that to my advantage. I also am a little unclear to what the parsing aspect means (I assumed it meant parsing through datatypes but I may be wrong) so I would love to have some more insight with that.*

*BONUS:*

* *You've identified some difficult parts. What are your thoughts on how you might solve them?*

The way I will solve these concerns I have will be through carefully reading through all lecture notes and pay 100% attention during the lecture classes. If I still struggle with any difficulties I will either email either my lab or lecture prof for clarification and use the internet to research solutions.

**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\_A11Spec\_S21 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]*

Algonquin College

Winter, 2022