Algonquin College Logo

# SCHOOL OF ADVANCED TECHNOLOGY

### ICT - Applications & Programming

### Computer Engineering Technology – Computing Science



A11

Language Specification

Lab Professor / Lab Session:

Paulo Sousa / 011

Team:

Henry Forget - Id: 041023812

Language Name [???]

|  |  |
| --- | --- |
| **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 ‘Sofia language to be Go compatible (see* <https://go.dev/>*).*
* *Option 2: Define a* ***DSL*** *– Proper to solve specific problems (ex: science, economy, music, etc.)..*

*This is going to be a 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 “Go-like” language or DSL]*

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

*[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 Go closely.]*

**Element 2 – Comments**

In Python, comments are declared with a hash (#) at the beginning of the line. \_\_\_ will opperate the same way, as follows:

# This is a comment

For multi-line comments, three quotation marks on either side of the text creates a multi-line comment in Python, \_\_\_ will again follow the same requirements:  
“ “ “

This

Is

Multi-line

“ “ “

**Element 3 – Keywords**

\_\_\_ contains many of the basic Python keywords which may not be used apart from that. All of their purposes are the same as they were originally used in Python.

**Element 4 – Datatypes**

Unlike C and many other programming languages, Python’s datatypes are infered and not declared. \_\_\_ will also follow this convention, however datatypes are still assigned behind the scenes thus it is still important to define them:

int: holds numbers between -2,147,483,647 to 2,147,483,647 as standard; 2 bytes

float: holds numbers between 1.2E-38 to 3.4E+38 as standard; 4 bytes

str:

bool: hold either ‘TRUE’ or ‘FALSE’; takes 1 byte of space

**Element 5 – Variables**

In Python, variables are simply declared with a name and a value, the datatype is infered. \_\_\_ will follow the same ruling, thus all three following lines are valid:

a = 123

b = “one, two, three”

c = 1.23

The data types of the variables created would be int, str and float in order.

**Element 6 – Methods / Functions**

Functions in \_\_\_\_ are defined by a preciding “def” keyword similarly to Python. Each line of code within the function must be indented by one tab stop.  
Something like:

def myfunction()  
 print(“Hello from function”)  
print(“Hello from outside”)

myfunction()

Would print the “Hello from outside” first as it’s not contained within the function.

Functions can have one, or several arguments contained within the brackets following the function name.

Something like:

def myfunction(name, age)  
 print(“My name is ” + name + “ and I am ” + age + “ years old!”)

myfunction(“Henry”, 21)

Would print “My name is Henry and I am 21 years old!”  
\* Variables can also be passed as parameters the same as any other programming language

**Element 7 - 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)?*
* ***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?*
* ***Interaction****: How will your code handle looping? (You can do one or more of a for-style loop, a while/do loop, etc.)*
* ***Input****: How does your program get input from the keyboard? (Strings are easiest.)*
* ***Output****: What would a programmer type to put output on the screen? What sort of variables or data will your code take?*
* ***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?*

**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.*

|  |  |
| --- | --- |
| **Part**  **2** | **Language Comparison** |

* No do loop
* No char
* Much bigger numbers
* Python can use quotes and double quotes

**Comparing with C language**

**Differences**

|  |  |  |
| --- | --- | --- |
|  | [Explanation] |  |

**Advantages / Disadvantages (in comparison with C)**

|  |  |  |
| --- | --- | --- |
|  | [Explanation] |  |

**Comparing with another language**

**Language Name:**

**Differences**

|  |  |  |
| --- | --- | --- |
|  | [Explanation] |  |

**Advantages / Disadvantages (in comparison with this second language)**

|  |  |  |
| --- | --- | --- |
|  | [Explanation] |  |

|  |  |
| --- | --- |
| **Part**  **3** | **Architectural Questions** |

**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.)]*

**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.*

***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.*

*[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?*

*[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?*

**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\_F23-A11\_AnswerTemplate*** *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 AI tool (ex: ChatGPT), report here, including the references used.* |

Algonquin College

Fall, 2023