Algonquin College Logo

# SCHOOL OF ADVANCED TECHNOLOGY

### ICT - Applications & Programming

### Computer Engineering Technology – Computing Science



A11

Language Specification

Team:

Andrew Lorimer - Id :041056170 / Solomon Thangthong - Id: 041023691

**Language Name [Tetanus]**

***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 ‘Mold language to be Rust compatible (see* [Rust](https://www.rust-lang.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**

Tetanus is a general purpose language compatible with rust. A tetanus code file will have a .tet file extension. This new language is inspired by both Java and Rust. This GPL aims to simplify and improve usability for users new to Rust that have experience developing in Java. By combining Rust and Java we aim to create an easy to use GPL with all the advantages Java and Rust have to offer.

**Element 2 – Comments**

Comments are made in tetanus the same way they are made in rust, using a double slash. Multi-line comments are made using a slash combined with asterisks.

//this is a comment

/\* this is a

multi line comment \*/

**Element 3 – Keywords**

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

final : replacement for the keyword “mut” in Rust to indicate a variable is mutable. By default all variables will be mutable in Tetanus.

function: replacement for the keyword “fn” in Rust to indicate a function is being declared.

while( ) : addition of parenthesis required for easier readability

class() : replacement for the use of structs in Rust to group and define objects

this. : replacement for the use of self. for Java users to reference

for, while, let, foreach, new, static, const, mut, true, false, return etc

**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]*

**integers** : are defined how they are in java using explicit key words (byte,short,int,long) the range for each signed datatype is as follows

byte: 1 Byte - 8 bits : Range [-128:127]

short: 2 Bytes -16 bits : Range [ -32768 : 32767 ]

int 4 Bytes - 32 bits : Range [ -2,147,483,648 : 2,147,483,547 ]

integers

**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.]*

**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)?*
* ***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** | **Examples** |
| --- | --- |

**Option 1: Rust-like**

**Hello World**

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

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

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

*[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.)]*

**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.*

*[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?*

**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]*

**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

Summer, 2023