## IS 214 - Assignment 1

INTERPOL Syntax Checker

### **Introduction**

INTERPOL, which stands for Integer Program Oriented Language, is a language specifically designed for IS 214 students. Compared to its IS 214 predecessors, INTERPOL is the simplest of them all. What makes it “integer oriented” is that the PL primarily deals with integers and consequently, no floating-point values are allowed. This implies that the results of arithmetic operations are automatically converted to integers. Strings are also allowed for printing purposes.

### **Objectives**

The goal of this programming assignment is to implement a syntax checker in Python for the INTERPOL programming language. Your program must be able to do the following:

1. Accept expressions from the user.
2. Determine whether the expression has syntax error or not.

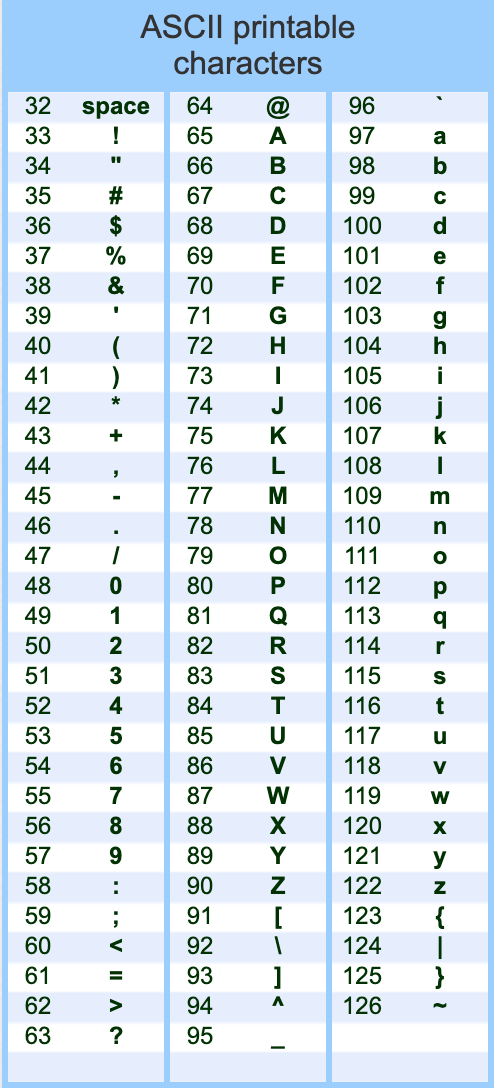
### **Formatting**

* Spaces are used to demarcate tokens in the language.
* Multiple spaces and tabs are treated as single spaces and are otherwise considered irrelevant.
* Indentation is also irrelevant.
* INTERPOL is **case sensitive**.

### **Syntactic Elements**

1. **Character Set:** set of symbols used in the programming language

Interpol is able to process all ASCII printable characters.



1. **Identifiers:** strings used to name variables, data objects, procedures/functions, etc.

* Should be less than 50 characters
* Should start with a character in the alphabet.
* Can be uppercase/lowercase

1. **Operator Symbols:** symbols used to represent the primitive operations in the language

Interpol does not use operator symbols. Instead, it uses the following keywords for the primitive operations.

|  |  |  |
| --- | --- | --- |
| **Operation** | **Keyword** | **Syntax** |
| Addition (num1 + num2) | PLUS | PLUS <num1> <num2> |
| Subtraction (num1 - num2) | MINUS | MINUS <num1> <num2> |
| Multiplication (num1 \* num2) | TIMES | TIMES <num1> <num2> |
| Division (num1 / num2) | DIVBY | DIVBY <num1> <num2> |
| Modulo (num1 % num2) | MODU | MODU <num1> <num2> |

1. **Keyword:** an identifier used as a fixed part of the syntax.

**Reserved words**: keywords that may not be used as a programmer-chosen identifier.

In INTERPOL, **all keywords are reserved words.**

The following are the keywords used by INTERPOL.

|  |  |  |
| --- | --- | --- |
| **Keyword** | **Syntax** | **Function** |
| CREATE | CREATE | Marks the beginning of the program. |
| GIVEYOU! | GIVEYOU! <string> | Prints the expression. |
| GIVEYOU!! | GIVEYOU!! <string> | Prints the expression with a new line at the end. |
| PLUS | PLUS <num1> <num2> | Adds num1 and num2 |
| MINUS | MINUS <num1> <num2> | Subtracts num2 from num1 |
| TIMES | TIMES <num1> <num2> | Multiplies num1 to num2 |
| DIVBY | DIVBY <num1> <num2> | Divides num1 by num2 |
| MODU | MODU <num1> <num2> | Gets the remainder of num1 divided by num2 |
| RUPTURE | RUPTURE | Marks the end of the program. |

1. **Comments:** words ignored during translation

All lines starting with # are considered comments.

1. **Delimiters:** used to mark the beginning or end of some syntactic constructs

In INTERPOL, a new line is a delimiter.

1. **Field Format**

INTERPOL uses a free-field format where program statements can be written anywhere on an input line without regard for positioning.

1. **Expressions**

In INTERPOL, expressions are string literals or numbers (integers). Strings should be enclosed in double quotes. " "

1. **Statement**

Using the correct syntax, each keyword in INTERPOL can be considered a statement.

1. **Overall Structure**

Programs should begin with the CREATE keyword and end with the RUPTURE keyword.

Sample INTERPOL Program:

CREATE

GIVEYOU!! 123

GIVEYOU! "Hello"

GIVEYOU!! ", world."

PLUS 1 2

TIMES 3 4

RUPTURE

In this assignment, you are to accept a line of input from the user and then tell the user if the syntax of his/her input is correct or not. See the following examples.

|  |  |
| --- | --- |
| INTERPOL Syntax Checker  Input CREATE to begin. Input RUPTURE to END.  $ CREATE  The syntax is correct. Beginning syntax checker.  $ GIVEYOU! "I have a pen"  The syntax is correct.  $ CREATE  The syntax is correct.  $ TIMES 3 4 5  The syntax is incorrect.  $ PRINT :)  The syntax is incorrect.  $ GIVEYOU! "リア"  The syntax is incorrect.  $ #Heyow  The syntax is correct.  $ RUPTURE  Thank you for using the syntax checker. | TIMES only has 2 arguments  PRINT is not a valid keyword  リア characters are not part of ASCII |

### **Submission**

* Make sure to use Python 3+!
* Name your file as first and (middlename initials-lastname)-01.py

Example: Since my name is Ria Mae H. Borromeo, the filename of my assignment should be ***rmh-borromeo-01.py***

* Submit your file in the submission bin