The SWING Project :

Prelude:
The SWING project is a programming language compiled and simulated on a virtual machine. It has a compiler written in C#, and a virtual machine written in DPL.
It allows you to create public numerical variables, data calculation, variable printing, goto and others.

Warning:
Read this information if you writing a program in SWING:
1. Be careful not to do a goto on an argument line, otherwise the virtual machine will not be able to read the program normally, because the virtual machine will try to read the arguments without knowing that there is a context. The goto lines only work on compiled SWING programs.
2. Do not leave blank lines in your SWING programs as this could create problems with conversions of numerical values.
3. You can create a maximum of 5 numeric variables in SWING.

Note: SWING programs are interpreted line by line, both functions and arguments.

Functions:

<PUBLIC> Function:
This function is used to create public numerical variables.
To declare the variables type between 1 and 5, to use them it is the same system.
OPCode: 300
Code:

```
PUBLIC: VAR_1 = NUMERICAL_VALUE
```

<GOTO> Function:
This function allows to execute a piece of code from a line, you can easily create a loop and functions with this function.
OPCode: 700
Code:

```
GOTO: PROGRAM_LINE
```

<REMARK> Function:
This function is used to create a comment in the code, it is not executed in the virtual machine.
OPCode: 0
Code:

```
REMARK: Hello, World!
```

<PRINT> Function:
This function is used to print a numerical variable.
Note: You can print variables and direct numerical values.
OPCode: 800
Code:

```
PRINT VAR_1
```

<RANDOM> Function:

This function is used to generate a random numerical value among an argument.
OPCode: 600
Code:

```
RND: VAR_1 # RANDOM_VALUE
```

<SLEEP> Function:

This function is used to pause the system for somes milliseconds.
OPCode: 50
Code:

```
SLEEP: MILLISECONDS_VAL
```

<IF> Function:

This function is used to execute a function only if a value is defined as 'true'.
If Operators:

(:) Operator:
Equal to…
OPCode: 220

(<) Operator:
Smaller than...
OPCode: 240

(>) Operator:
Bigger than...
OPCode: 260

(!) Operator:
Is not...
OPCode: 280

And you can use 'EX' function to terminate 'IF' function.
Code:

```
IF VAR_1 IF_OP VAR_2
...
EX
```

<PAUSE> Function:

This function is used to set system in pause.
OPCode: 30
Code:

```
PAUSE
```

<INPUT> Function:
This function is used to input a numerical value in a variable.
OPCode: 400
Code:

```
INPUT: VAR_1
```

<CALC> Function:
This function is used to create a numerical calculation with the five operators of the SWING
Note: To create a numeric calculation with this function, you are required to use variables and not direct numerical values.
And you can use the character <#> to pass the computed value to a variable declared or not.
OPCode: 900
Code:

```
CALC: VAR_1 + VAR_2 # VAR_3
```

<END> Function:
This function is used to terminate any SWING program.
OPCode: 500
Code:

```
...
END
```

Operators:

<+> Operator:
Add.
OPCode: 120

<-> Operator:
Minus.
OPCode: 140

<*> Operator:
Multiply.
OPCode: 160

</> Operator:
Divide.
OPCode: 180

<%> Operator:
Modulo.
OPCode: 200

Samples:

<Hello-World>

```
REMARK: SWING | Hello, World!
PUBLIC: 1 = 68
PUBLIC: 2 = 80
PUBLIC: 3 = 76
PUBLIC: 4 = 33
PRINT 1
PRINT 2
PRINT 3
PRINT 4
END
```

<Counting-Program>

```
REMARK: SWING | Counting Program
PUBLIC: 1 = 1
PUBLIC: 2 = 1
PRINT 1
CALCU: 1 + 2 # 1
PRINT 1
GOTO: 11
END
```

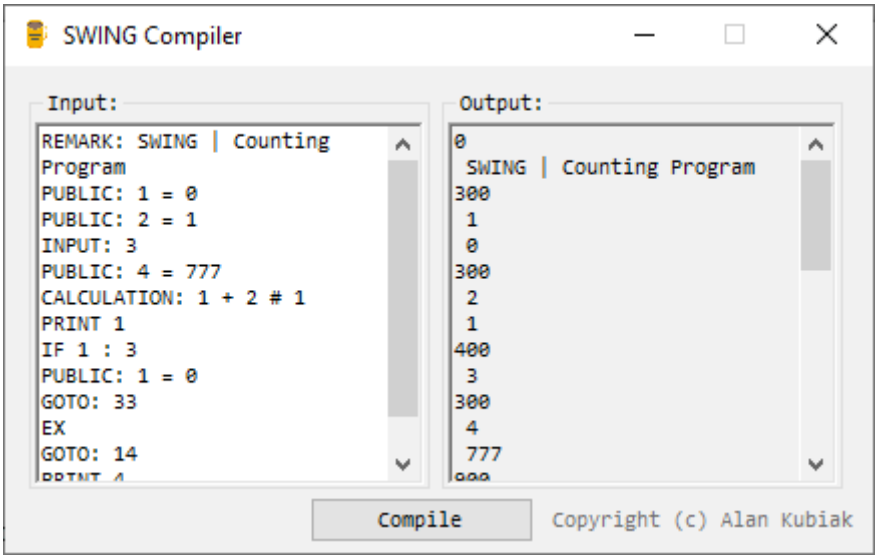
Compiled Program:

```
0 ‘Comment
 SWING | Counting Program
300 ‘Variable declaration
 1
 1
300 ‘Variable declaration
 2
 1
800 ‘PRINT
 1
900 ‘CALCULATION
 1
120
 2  ‘Arguments
 1
800 ‘PRINT
 1
700 ‘GOTO
 11
500 ‘END
```

How to code ?:

You can start coding your first SWING program by downloading the SWING development pack from this address:
https://ubik.tk/swing/SWING-DK-061821.zip

1. Open <Swing Compiler.exe>.



You can enter your program in the "Input:" box and click on the "Compile" button to compile the program.

2. Copy the compiled program from the "Output:" box, you can now implement your compiled program in the "program.swng" file by editing it with text editing software like Notepad++ or Sublime Text and save it in the same directory as the executable "swing.exe".

3. Run the "swing.exe" program and your "program.swng" program will be executed.

Final Note: This project is simulated on the DPL virtual machine written in C++ (DPLVM), and this project was created to test the technical capabilities of the DPL programming language, thanks for reading.

-Alan Kubiak