CS301
Assignment #1
Thomas Crow

Pascal

Assignments > Assignment #1 > P ForLoop.pas

```pascal
// Assignment#1-- Pascal
// Thomas Crow
//
// Using JavaScript (not Java, JavaScript is a different language),
// Rust, and Pascal write a simple FOR loop that adds the first 100
// positive integers (from 1 to 100, included).
// Print the result of that summation.
// Use proper comments in your programs.


program ForLoop;

  var
    sum,i: integer;

begin
  //Initialize variables
  sum := 0;
  i := 0;

  //Add first 100 positive integers, assgn to value sum
  for i := 1 to 100 do
    sum := sum + i;

  //Print the value of sum
  writeln('The sum is ',sum);

end.
```

```
tbcrow@DigDug Assignment #1 % ls
ForLoop.js      ForLoop.pas     ForLoop.rs
tbcrow@DigDug Assignment #1 % fpc ForLoop.pas
Free Pascal Compiler version 3.2.2 [2021/05/16] for aarch64
Copyright (c) 1993-2021 by Florian Klaempfl and others
Target OS: Darwin for AArch64
Compiling ForLoop.pas
Assembling forloop
Linking ForLoop
26 lines compiled, 0.6 sec
tbcrow@DigDug Assignment #1 % ls
ForLoop         ForLoop.js      ForLoop.o       ForLoop.pas     ForLoop.rs
tbcrow@DigDug Assignment #1 % ./ForLoop
The sum is 5050
tbcrow@DigDug Assignment #1 %
```

Javascript

ForLoop.js U | ForLoop.pas U

Assignments > Assignment #1 > ForLoop.js > ...

```javascript
// Assignment #1 -- Javascript
// Thomas Crow
//
// Using JavaScript (not Java, JavaScript is a different language),
// Rust, and Pascal write a simple FOR loop that adds the first 100
// positive integers (from 1 to 100, included).
// Print the result of that summation.
// Use proper comments in your programs.

//initialize values
let sum = 0;

//Add first 100 positive integers, assgn to value sum
for (let i = 0; i < 100; i++, sum += i); //Add positive integers 1 to 100 and assign to i

//Print the value of sum
console.log("The sum is " + sum);
```

```
tbcrow@DigDug Assignment #1 % ls
ForLoop.js      ForLoop.pas     ForLoop.rs
tbcrow@DigDug Assignment #1 % node ForLoop.js
The sum is 5050
tbcrow@DigDug Assignment #1 %
```

Rust

Conclusion:

While I think it's hard to gain too much insight into each program languages characteristics based on such trivial coding, I think some characteristics are present. I think Rust is the easiest language to read and understand. At least for programmers used to languages like Java, Python, and C++. I suppose if the programmer was more used to languages like FORTRAN and COBOL, Pascal would seem more natural. As it was for me going from BASIC to FORTAN in high school. However, that will be a very small percentage of current programmers indeed. I am not very familiar with JavaScript. However, the use of the console.log command to print output to the screen did not feel natural at all. Granted the sample size is very small.

While Pascal was written with the intention of teaching good programming form, this ideal form is not applicable to programming languages designed for Object Oriented Programming. The learning curve would be steep for most coming from modern languages. Or even those who learned it in high school and had to try and remember it! With both Rust and JavaScript embracing OOP, I'm not sure which would have a harder learning curve. Though given the limited experience with both, I think Rust would have more similarities for those coming from again C++, Java, and Python.

Given the rather ancient support for Pascal, limited primarily to learning purposes, I cannot see Pascal having anything near considered competitive troubleshooting tools. Such as debuggers and testing tools. My guess would be that Rust would have a more robust troubleshooting toolset, given it's more traditional programming pedigree. However, with JavaScript being so ubiquitous in web programming, perhaps the amount of troubleshooting is comparable or even exceeds Rust.