[[1]](#footnote-1)

Assignment 1 – Python-like interpreter

Yvan Gihoza

*Overview*—As our first assignment we were to design a python-like interpreter. Our program had to meet some expected operations and had to pass some test. The interpreter had to parse the command string into three different operations being the assignment operation, print operation and the list appending. Accepted variables were to be long integer, double floating point, character, string and list.

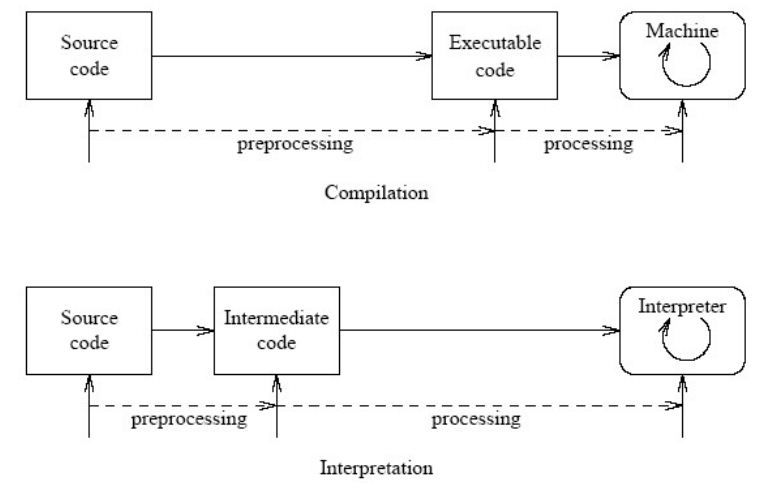
# INTRODUCTION

S

Ome of the basic operations which had to be met where the arithmetic operations like, +, -, \*and /. Where the program had to start with “>>>” prompting the user for input and then that input had to be parsed using the parser provided and perform the different operations.

# Application Design

I started by reviewing the difference between a compiler and an interpreter, after understanding how the interpreter works that is when I decided to get started on the project.



## Recommendations

I then went back to the code to understand how the parser provided works, so I could easily implement my program.

# Results

Since there was given some basic operations which had to be met, the testing part was just plugging in the, testing tools provided like saying >>>a=2

Print(a): 2 and so on. Here are some of the testing that I did.

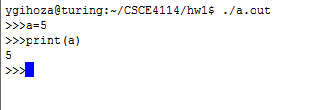
# Conclusion

This was my first big project in C language and to be honest it was really challenging but thanks to the TA he was of big help for me to better understand it and design it even though I was not able to finish.

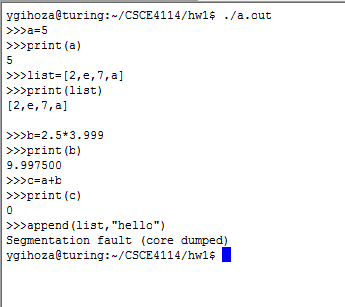
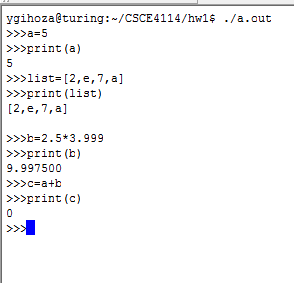
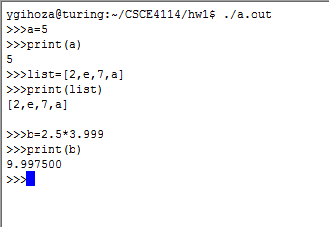
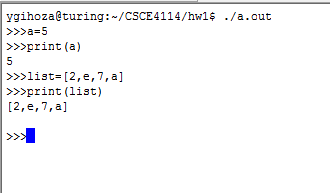
# Testing

I was not able to perform all the testing,

>>> a = 5



I was able to implement all the arithmetic functions, but I had trouble with the “append” command as I was having a segmentation fault and I was not able to resolve it.



1. [↑](#footnote-ref-1)