Andrew Vo
Professor Chinua Umoja
Test 2
November 16, 2020

Written Response

Question 1

Using regex I was able to implement an algorithm that reads each string, checks to see if it returns true for the regex rule it was compared to, and outputs the original lexeme along with its assigned token. For example, the lexeme $pearl_id would be recognized by the prefix '$' and a single '_' followed by characters. It would then be converted to the token "Perl". If the string did not fit any of the rules it was assigned Invalid.

Question 2

The array is bound to memory cells before program execution begins and remains unbound to those memory cells, so it makes sense that it does not take long for the Static function to execute. Compared to Heap which took much longer as heap is a disorganized collection of storage cells, and a variable allocated from the heap can only be referenced through a pointer. This is why creating arrays of the same size from the heap requires a significant amount of time compared to the static and stack array.

Question 3

<stmt> → <op> <var> <stmt>
<stmt> → <op> <var> <var>
<letter> → [a-z|A-Z]{<num>}
<var> → [<letter>|<num>]
<op> → [‘+’|‘-’|‘/’|‘*’|‘%’]
<num> → [0-9]{<num>}

Any bottom-up parser would suffice as the process of a top-down parser starts with the last sentential form and produces the sequence of sentential form until all that remains is the start symbol. For example, A+B/C*D \ would be read in preorder form which is +AB/*CD and from there would be parsed top-down.

Question 4

Static scoping which was introduced in ALGOL 60 is the method of binding names to nonlocal variables. It is named that because the scope of a variable can be statically determined prior to the execution. Dynamic scoping only occurs if the first binding occurs during runtime and could change in the course of program execution. With dynamic type binding it would not matter for

both and dynamic scoping because the symbol table would be different for both, so the type of scoping would not change the answer.

Question 5

I would implement a list composed of keywords in which I would implement a function that would read a file and compare the strings and character values to the list in order to analyze and assign tokens.I would allow users to add their own reserved words to the list. Nothing would change in the code besides that added function.

Question 6

Java while statement ENBF
<while_stmt> → "while" "(" <bool_stmt> ")" "{" <block> "}"

Java if Statement EBNF
<if_stmt> → if (<bool_expr>) <statement> [else <statement>]

Logical/Mathematical expression EBNF
<assign> → <var> "=" [<var> | <const> | <func> | "null"]

Mathematical assignment statement EBNF
<stmt> → <op> <var> [<stmt> | <var>]
<letter> → [a-z|A-Z]{<num>}
<var> → [<letter>|<num>]
<op> → [‘+’|‘-’|‘/’|‘*’|‘%’]
<num> → [0-9]{<num>}

Question 7

Beginning with a statement, if the token is assi code it would be checked for var or function. If it is an if statement it would be checked for open parenthesis, a boolean expression, closed parenthesis, open brackets, a block, blocked brackets and then a potential else or else if statement in which case it would cycle back through again. If it is a while statement, it would be checked for open parenthesis, a boolean expression, closed parenthesis, open brackets, a bock statement, and then closed backeets.

Question 8

In static scoping the variable always refers to its top level environment, which is just a property of the program text. While in dynamic scoping a global identifier refers to the identifier associated with the most recent environment. This means that every occurrence of an identifier is searched in the most recent binding. Perl's keyword "my" defines a statically scoped variable, while the keyword "local" defines a dynamically scoped variable.[1]

IEE Citation
1. “Static and Dynamic Scoping,” GeeksforGeeks, 22-Oct-2019. [Online]. Available: https://www.geeksforgeeks.org/static-and-dynamic-scoping/. [Accessed: 16-Nov-2020].