Sindhu Parajuli
1001764819

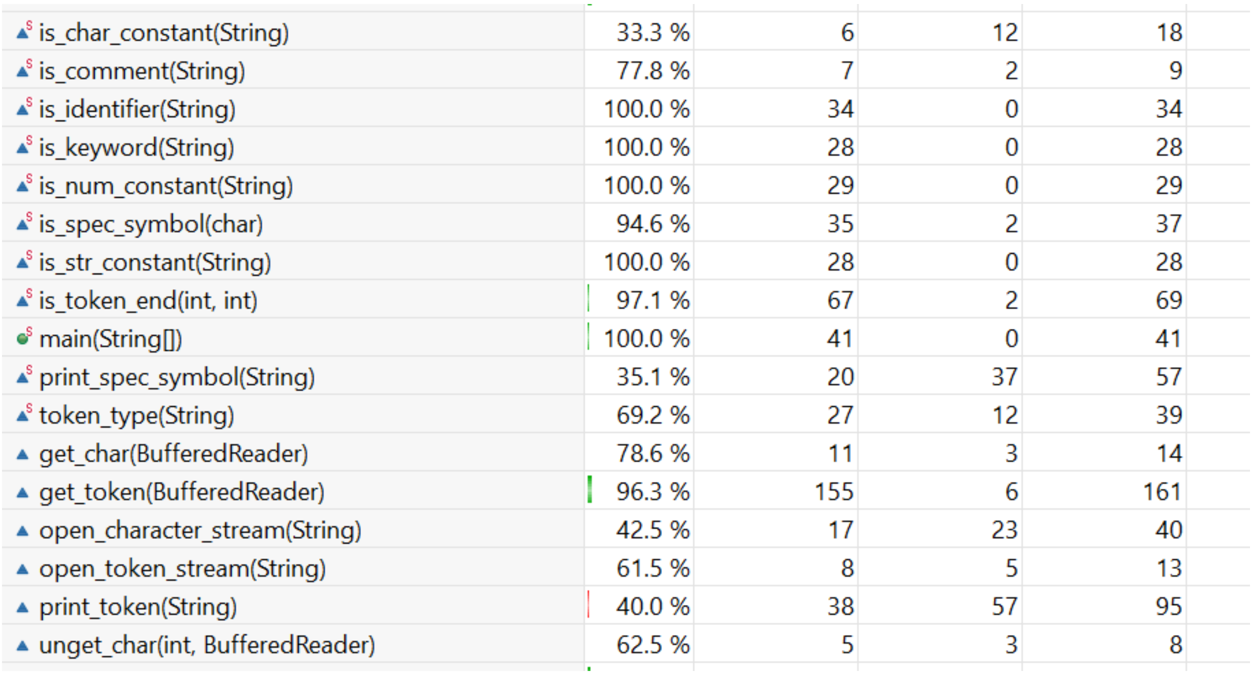
Summary & Discussion

The project for CSE 4321 (Software Testing) provided the practical side of the materials covered throughout the semester. The Project involved concepts from the beginning of the course work including dividing code into blocks, drawing CFGs, finding test-paths to creating test cases and writing JUnit tests to achieve those test coverages. Faults were detected with the help of test-cases and formed. 14 faults were detected during the course of the project.

The first portion of the project was creating CFGs. In order to create CFGs codes were analyzed and put into a block table, where the entry and exit points of the blocks were noted. The function calls were placed in separate blocks as instructed. Analysing the entry, exit points, the blocks were converted into a CFG using the online drawing tool ‘draw.io’. CFGs for 15 functions were created including the main function.

After the completion of CFGs, test paths were derived. All possible test-paths were written down. Now according to the test-paths, input and output were detected and noted. The GUI jar provided was very helpful in analysing the inputs and possible outputs. The completion of test-paths involved looking up the block table and corresponding lines of code to analyze the flow of the process.

After the identification of possible test-paths, using the detected inputs and possible outputs, unit tests were written for 7 functions including the main function. JUnit test yielded results such as follows:



is_char_constant(String)	33.3 %	6	12	18
is_comment(String)	77.8 %	7	2	9
is_identifier(String)	100.0 %	34	0	34
is_keyword(String)	100.0 %	28	0	28
is_num_constant(String)	100.0 %	29	0	29
is_spec_symbol(char)	94.6 %	35	2	37
is_str_constant(String)	100.0 %	28	0	28
is_token_end(int, int)	97.1 %	67	2	69
main(String[])	100.0 %	41	0	41
print_spec_symbol(String)	35.1 %	20	37	57
token_type(String)	69.2 %	27	12	39
get_char(BufferedReader)	78.6 %	11	3	14
get_token(BufferedReader)	96.3 %	155	6	161
open_character_stream(String)	42.5 %	17	23	40
open_token_stream(String)	61.5 %	8	5	13
print_token(String)	40.0 %	38	57	95
unget_char(int, BufferedReader)	62.5 %	5	3	8

After the completion of Unit testing, JaCoCo was used for overall branch coverage reports for the project. This gave an overview about the branches covered using the test-cases.

While executing the test-cases, faults were detected which were taken note of and placed on the ‘Fault detection’ part of the assignment. Eclipse IDE was used for the project as recommended.

Overall, the project provided an idea of Unit testing, possible faults on code and debugging for problems.