Test Case 1: (< 10 lines)

Code (6 lines):

@@

int a,b,c;

@@

if(a<b)

{a:= c;}

endif

Terminal Run:

Arts-MBP:ASGN3\_OC Arty$ python3.4 OC.py

Enter file you would like to open (type "quit" to exit): testcase1.txt

Lexer working...

File open!

...Lexer complete!

Your Tokens and Lexemes have been saved as testcase1.RAT in the working directory.

Object Code Generator running...

...Object Code Generator finished!

There were no errors!

Your syntactic analysis of testcase1 has been saved as testcase1.SA in the working directory.

Your object code of testcase1 has been saved as testcase1.OC in the working directory.

Would you like to process another file? (yes/no): no

Goodbye!

Contents of testcase1.OC:

Assembly Code

====================

1 PUSHM 1000

2 PUSHM 1001

3 LES

4 JUMPZ 7

5 PUSHM 1002

6 POPM 1000

Symbol Table

======================================

identifier Memory Location Type

a 1000 int

b 1001 int

c 1002 int

Test Case 2: (< 20 lines)

Code (12 lines):

@@

int i, max, sum;

@@

sum := 0;

i := 1;

read( max);

while(i< max){

sum:= sum +1;

i := i + 1;

}

max := 3;

write(sum+max);

Terminal Run:

Arts-MBP:ASGN3\_OC Arty$ python3.4 OC.py

Enter file you would like to open (type "quit" to exit): testcase2.txt

Lexer working...

File open!

...Lexer complete!

Your Tokens and Lexemes have been saved as testcase2.RAT in the working directory.

Object Code Generator running...

...Object Code Generator finished!

There were no errors!

Your syntactic analysis of testcase1 has been saved as testcase2.SA in the working directory.

Your object code of testcase1 has been saved as testcase2.OC in the working directory.

Would you like to process another file? (yes/no): no

Goodbye!

Contents of testcase2.OC:

Assembly Code

====================

1 PUSHI 0

2 POPM 1002

3 PUSHI 1

4 POPM 1000

5 STDIN

6 LABEL

7 PUSHM 1000

8 PUSHM 1001

9 LES

10 JUMPZ 20

11 PUSHM 1002

12 PUSHI 1

13 ADD

14 POPM 1002

15 PUSHM 1000

16 PUSHI 1

17 ADD

18 POPM 1000

19 JUMP 1001

20 PUSHI 3

21 POPM 1001

22 PUSHM 1002

23 PUSHM 1001

24 ADD

25 STDOUT

Symbol Table

======================================

identifier Memory Location Type

i 1000 int

max 1001 int

sum 1002 int

Test Case 3: (> 20 lines)

Code (27 lines):

@@

int i, max, sum;

boolean x;

@@

sum := 0;

i := 1 + 2;

read( max);

while(i < max)

{

sum:= sum +1;

i := i + 1;

}

max := 3;

write(sum+max);

if (max < sum)

{

x := 1;

} else

{

x := 0;

}endif

Terminal Run:

Arts-MBP:ASGN3\_OC Arty$ python3.4 OC.py

Enter file you would like to open (type "quit" to exit): testcase3.txt

Lexer working...

File open!

...Lexer complete!

Your Tokens and Lexemes have been saved as testcase3.RAT in the working directory.

Object Code Generator running...

...Object Code Generator finished!

There were no errors!

Your syntactic analysis of testcase1 has been saved as testcase3.SA in the working directory.

Your object code of testcase1 has been saved as testcase3.OC in the working directory.

Would you like to process another file? (yes/no): no

Goodbye!

Contents of testcase3.OC:

Assembly Code

====================

1 PUSHI 0

2 POPM 1002

3 PUSHI 1

4 PUSHI 2

5 ADD

6 POPM 1000

7 STDIN

8 LABEL

9 PUSHM 1000

10 PUSHM 1001

11 LES

12 JUMPZ 22

13 PUSHM 1002

14 PUSHI 1

15 ADD

16 POPM 1002

17 PUSHM 1000

18 PUSHI 1

19 ADD

20 POPM 1000

21 JUMP 1001

22 PUSHI 3

23 POPM 1001

24 PUSHM 1002

25 PUSHM 1001

26 ADD

27 STDOUT

28 PUSHM 1001

29 PUSHM 1002

30 LES

31 JUMPZ 35

32 PUSHI 1

33 POPM 1003

34 JUMP 37

35 PUSHI 0

36 POPM 1003

Symbol Table

======================================

identifier Memory Location Type

i 1000 int

max 1001 int

sum 1002 int

x 1003 boolean