EXPERIMENT 15

AIM

Implement the back end of the compiler which takes the three address code and produces the 8086 assembly language instructions that can be assembled and run using an 8086 assembler. The target assembly instructions can be simple move, add, sub, jump etc.

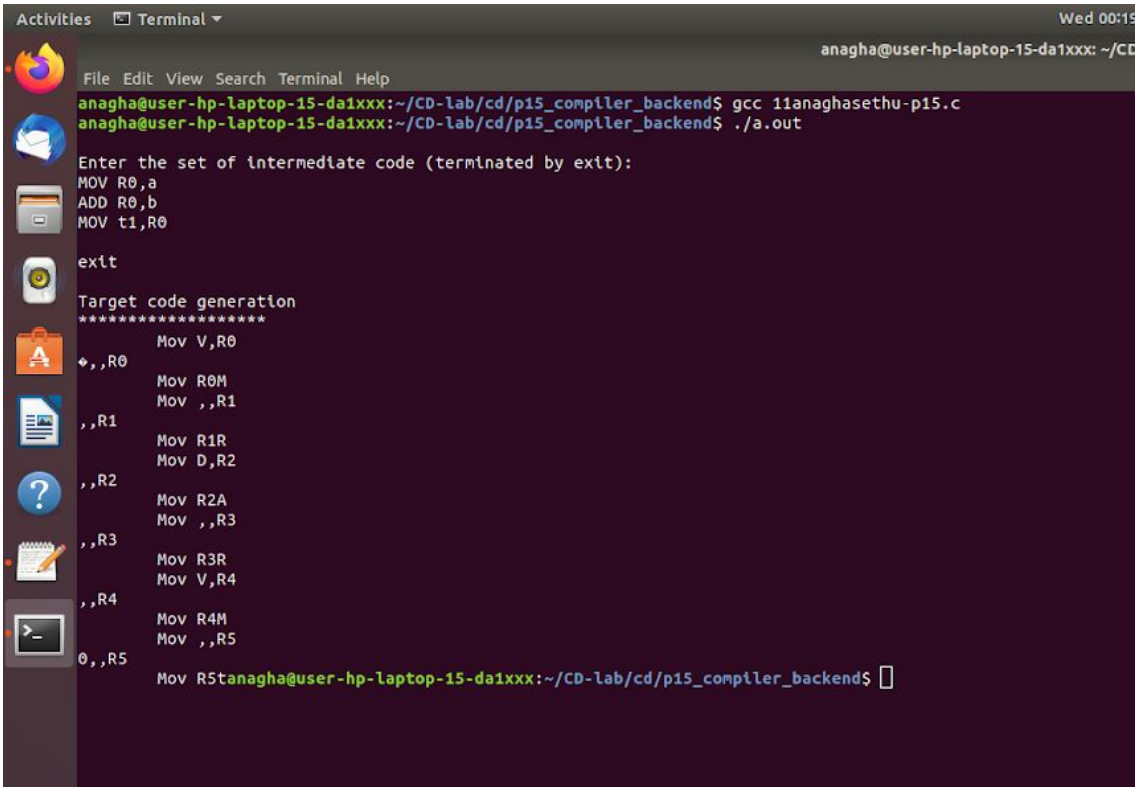
ALGORITHM

1. Start
2. Open the source file and store the contents as quadruples.
3. Check for operators, in quadruples, if it is an arithmetic operator generator it or if assignment operator generates it, else perform unary minus on register C.
4. Write the generated code to output definition of the file.
5. Print the output.
6. Stop

OUTPUT

```
gcc 11anaghasethu-p15.c

./a.out
```


Activities
Terminal
Wed 00:19
anagha@user-hp-laptop-15-da1xxx: ~/CD
File Edit View Search Terminal Help
anagha@user-hp-laptop-15-da1xxx:~/CD-lab/cd/p15_compiler_backend$ gcc 11anaghasethu-p15.c
anagha@user-hp-laptop-15-da1xxx:~/CD-lab/cd/p15_compiler_backend$ ./a.out
Enter the set of intermediate code (terminated by exit):
MOV R0,a
ADD R0,b
MOV t1,R0
exit
Target code generation
****
Mov V,R0
◆,,R0
Mov R0M
Mov ,,R1
,,R1
Mov R1R
Mov D,R2
,,R2
Mov R2A
Mov ,,R3
,,R3
Mov R3R
Mov V,R4
,,R4
Mov R4M
Mov ,,R5
0,,R5
Mov R5tanagha@user-hp-laptop-15-da1xxx:~/CD-lab/cd/p15_compiler_backend$