

 **Ex-6-CD-Lab** Public

forked from [RamachandranSEC/Ex-6-CD-Lab](#)

 1 Branch

 Tags






**About** **Add file** 






**Code** 

No description, website, or topics provided.

 **ganeshprabhu2005** Update README.md

4944a8b · now 

 README.md

-  Readme
-  Activity
-  0 stars
-  0 watching
-  0 forks

## Releases

No releases published  
[Create a new release](#)

## Packages

No packages published  
[Publish your first package](#)

# Ex-6-IMPLEMENTATION-OF-THE-BACK-END-OF-THE-COMPILER-

IMPLEMENTATION OF THE BACK END OF THE COMPILER

Date :

Aim :

To write a program to implement the back end of the compiler.

## ALGORITHM

1. Start the program.
2. Get the three variables from statements and stored in the text file k.txt.

3. Compile the program and give the path of the source file.
4. Execute the program.
5. Target code for the given statement is produced.
6. Stop the program.

## PROGRAM

```
#include <stdio.h>
#include <ctype.h>
#include <stdlib.h>
int main()
{
    int i = 2, j = 0, k = 2, k1 = 0;
    char ip[10], kk[10];
    FILE *fp;
    printf("Enter the filename of the intermediate
code: ");
    scanf("%s", kk);
    fp = fopen(kk, "r");
    if (fp == NULL) {
        printf("\nError in opening the file\n");
        return 1;
    }
    printf("\nStatement\tTarget Code\n\n");
    while (fscanf(fp, "%s", ip) != EOF)
    {
        printf("%s\tMOV %c,R%d SUB ", ip, ip[i + k],
j);
        if (ip[i + 1] == '+')
            printf("ADD ");
        else
            printf("SUB ");
        if (islower(ip[i]))
            printf("%c,R%d\n", ip[i + k1], j);
        else
            printf("%c,%c\n", ip[i], ip[i + 2]);
        j++;
        k1 = 2;
        k = 0;
    }
}
```

📖 README

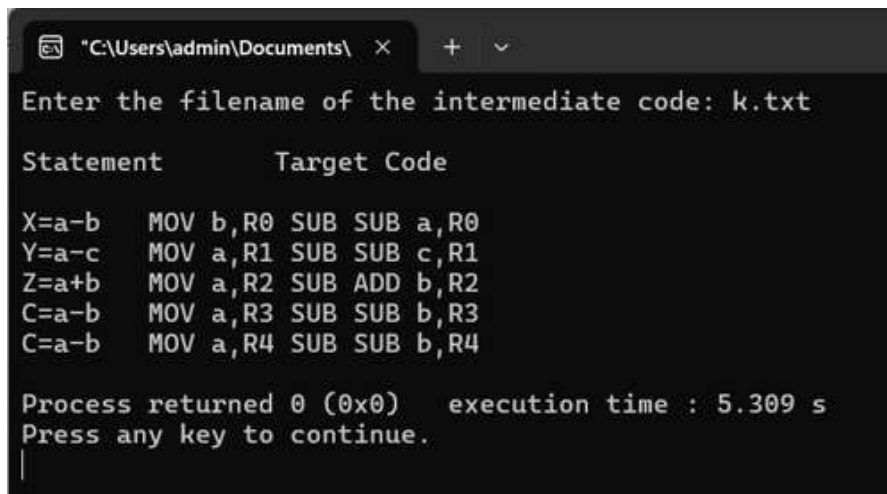


## OUTPUT



```
k.txt
File Edit View

X=a-b
Y=a-c
Z=a+b
C=a-b
C=a-b
```



```
"C:\Users\admin\Documents\ k.txt + v
Enter the filename of the intermediate code: k.txt

Statement      Target Code
X=a-b          MOV b,R0 SUB SUB a,R0
Y=a-c          MOV a,R1 SUB SUB c,R1
Z=a+b          MOV a,R2 SUB ADD b,R2
C=a-b          MOV a,R3 SUB SUB b,R3
C=a-b          MOV a,R4 SUB SUB b,R4

Process returned 0 (0x0)  execution time : 5.309 s
Press any key to continue.
|
```

## Result

The back end of the compiler is implemented successfully, and the