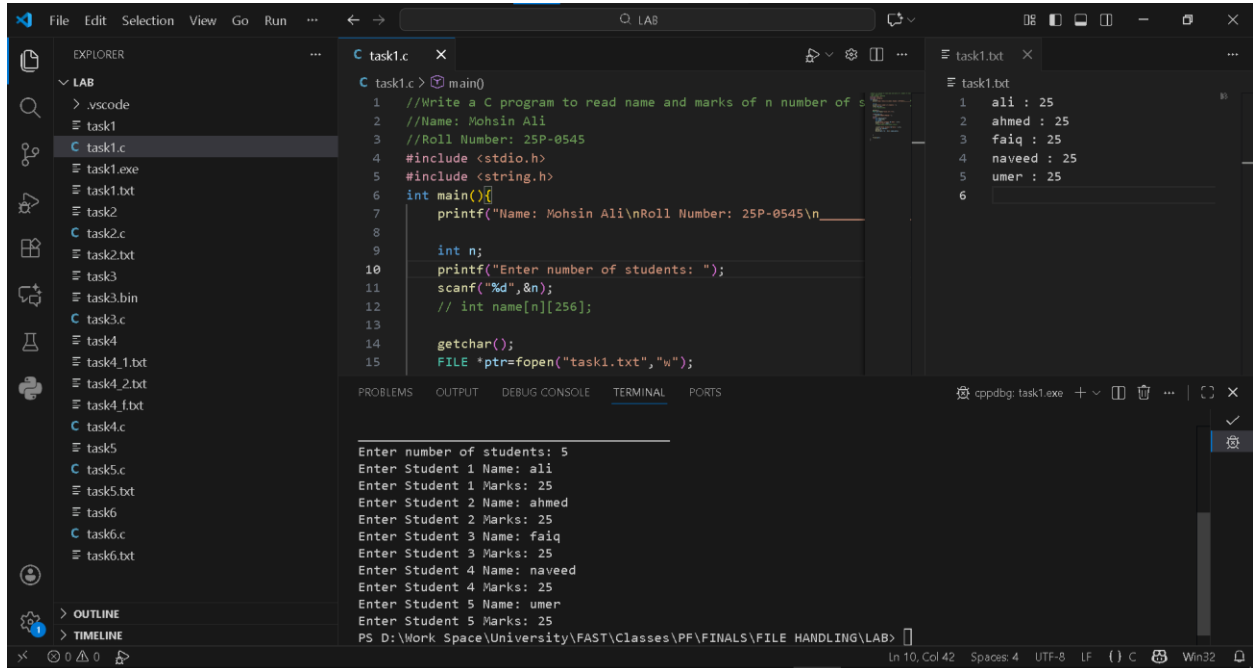




PROGRAMMING FUNDAMENTALS - LAB

Name: Mohsin Ali
Roll Number: 25P-0545

Q#1: Write a C program to read name and marks of n number of students and store them in a file



The screenshot shows a Visual Studio Code editor with a C program in `task1.c` and its output in `task1.txt`. The program reads the user's name and roll number, then asks for the number of students. It then loops to read each student's name and marks, storing them in a file named `task1.txt`.

```
1 //Write a C program to read name and marks of n number of s
2 //Name: Mohsin Ali
3 //Roll Number: 25P-0545
4 #include <stdio.h>
5 #include <string.h>
6 int main()
7 {
8     printf("Name: Mohsin Ali\nRoll Number: 25P-0545\n");
9
10    int n;
11    printf("Enter number of students: ");
12    scanf("%d",&n);
13    // int name[n][256];
14
15    getchar();
16    FILE *ptr=fopen("task1.txt","w");
```

The output file `task1.txt` contains the following data:

```
1 ali : 25
2 ahmed : 25
3 faiq : 25
4 naved : 25
5 umer : 25
6
```

CODE:

```
//Write a C program to read name and marks of n number of students and store them in a file
```

```
//Name: Mohsin Ali
```

```
//Roll Number: 25P-0545
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main(){
```

```
    printf("Name: Mohsin Ali\nRoll Number: 25P-0545\n_____\\n");
```

```
    int n;
```

```
    printf("Enter number of students: ");
```

```
    scanf("%d",&n);
```

```
// int name[n][256];

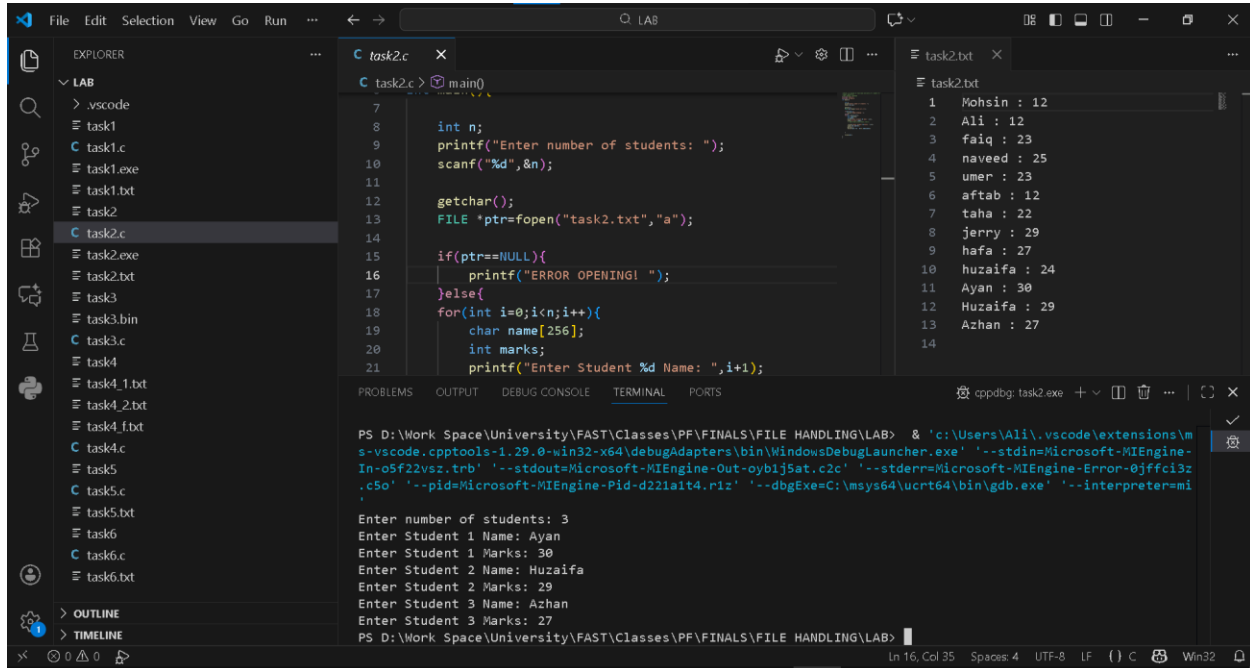
getchar();

FILE *ptr=fopen("task1.txt","w");

if(ptr==NULL){
    printf("ERROR OPENING! ");
}else{
    for(int i=0;i<n;i++){
        char name[256];
        int marks;
        printf("Enter Student %d Name: ",i+1);
        scanf("%[^\n]s",name);
        // fgets(name,sizeof(name),stdin);

        printf("Enter Student %d Marks: ",i+1);
        scanf("%d",&marks);
        getchar();
        fprintf(ptr,"%s : %d\n",name,marks);
    }
    fclose(ptr);
}
}
```

Q#2: Write C program to read name and marks of n number of students from and store them in a file. If the file previously exists, add the information to the file.



The screenshot shows a Visual Studio Code editor with a C program named `task2.c` and its output in the terminal. The program prompts the user to enter the number of students, then for each student, it prompts for the name and marks, and finally stores this information in a file named `task2.txt`. The terminal output shows the program running successfully, with the user entering 3 students and their details being stored in the file.

```
task2.c
7
8     int n;
9     printf("Enter number of students: ");
10    scanf("%d",&n);
11
12    getchar();
13    FILE *ptr=fopen("task2.txt","a");
14
15    if(ptr==NULL){
16        printf("ERROR OPENING! ");
17    }else{
18        for(int i=0;i<n;i++){
19            char name[256];
20            int marks;
21            printf("Enter Student %d Name: ",i+1);
```

```
task2.txt
1 Mohsin : 12
2 Ali : 12
3 faiq : 23
4 naved : 25
5 umer : 23
6 aftab : 12
7 taha : 22
8 jerry : 29
9 hafa : 27
10 huzaifa : 24
11 Ayan : 30
12 Huzaifa : 29
13 Azhan : 27
14
```

```
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB> & 'c:\Users\Ali\.vscode\extensions\ms-vscode.cpptools-1.29.0-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-o5f22vsz.trb' '--stdout=Microsoft-MIEngine-Out-oybj5at.c2c' '--stderr=Microsoft-MIEngine-Error-0jffc13z.c5o' '--pid=Microsoft-MIEngine-Pid-d221ait4.r1z' '--dbgExe=C:\msys64\bin\gdb.exe' '--interpreter=mi'
Enter number of students: 3
Enter Student 1 Name: Ayan
Enter Student 1 Marks: 30
Enter Student 2 Name: Huzaifa
Enter Student 2 Marks: 29
Enter Student 3 Name: Azhan
Enter Student 3 Marks: 27
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB>
```

CODE:

//Write a C program to read name and marks of n number of students and store them in a file

//Name: Mohsin Ali

//Roll Number: 25P-0545

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main(){
```

```
    int n;
```

```
    printf("Enter number of students: ");
```

```
    scanf("%d",&n);
```

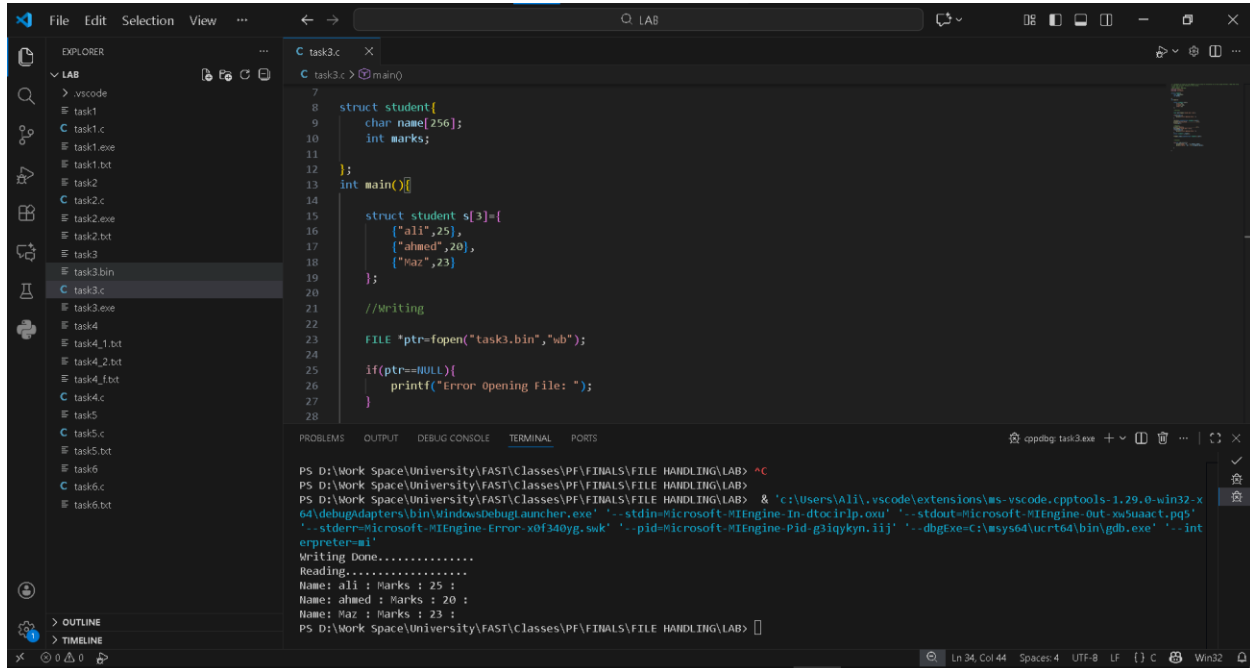
```
    getchar();
```

```
    FILE *ptr=fopen("task2.txt","a");
```

```
    if(ptr==NULL){
```

```
    printf("ERROR OPENING! ");
}else{
for(int i=0;i<n;i++){
    char name[256];
    int marks;
    printf("Enter Student %d Name: ",i+1);
    scanf("%[^\n]s",name);
    // fgets(name,sizeof(name),stdin);
    printf("Enter Student %d Marks: ",i+1);
    scanf("%d",&marks);
    getchar();
    fprintf(ptr,"%s : %d\n",name,marks);
}
fclose(ptr);
}
}
```

Q#3: C program to write all the members of an array of structures to a file using fwrite(). Read the array from the file and display on the screen.



The screenshot shows a VS Code editor with a C program named `task3.c` open. The program defines a `struct student` with `char name[256]` and `int marks`. In the `main` function, an array `s` of three `student` structures is initialized with the following data:

```
struct student s[3]={
    {"ali",25},
    {"ahmed",20},
    {"kaz",23}
};
```

The program then attempts to write this array to a file named `task3.bin` using `fwrite`. The terminal output shows the program running successfully and displaying the contents of the array:

```
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB> ^C
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB>
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB> & 'c:\Users\Alii\.vscode\extensions\ms-vscode.cpptools-1.29.0-win32-x64\debugadapter\bin\windowsdebuglauncher.exe' '--stdin=Microsoft-MIEngine-in-dto:rlp.ou' '--stdout=Microsoft-MIEngine-out-..Suact.pg5'
'--stderr=Microsoft-MIEngine-Error-xbf340yg.swk' '--pid=Microsoft-MIEngine-Pid-g3iqykyn.iiij' '--dbgExe=c:\wsys64\ucrtd\bin\gdb.exe' '--int
erpreter=mi'
Writing Done.....
Reading.....
Name: ali : Marks : 25 :
Name: ahmed : Marks : 20 :
Name: kaz : Marks : 23 :
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB>
```

CODE:

```
// C program to write all the members of an array of structures to a file using fwrite(). Read the array
```

```
// from the file and display on the screen.
```

```
//Name: Mohsin Ali
```

```
//Roll Number: 25P-0545
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
struct student{
```

```
    char name[256];
```

```
    int marks;
```

```
};
```

```
int main(){
```

```
    struct student s[3]={
```

```
        {"ali",25},
```

```
        {"ahmed",20},
```

```
        {"Maz",23}
```

```
    };
```

```
    //Writing
```

```
    FILE *ptr=fopen("task3.bin","wb");
```

```
    if(ptr==NULL){
```

```
        printf("Error Opening File: ");
```

```
    }
```

```
    fwrite(s,sizeof(struct student),3,ptr);
```

```
    printf("Writing Done.....\n");
```

```
    fclose(ptr);
```

```
    //Reading
```

```
    printf("Reading.....\n");
```

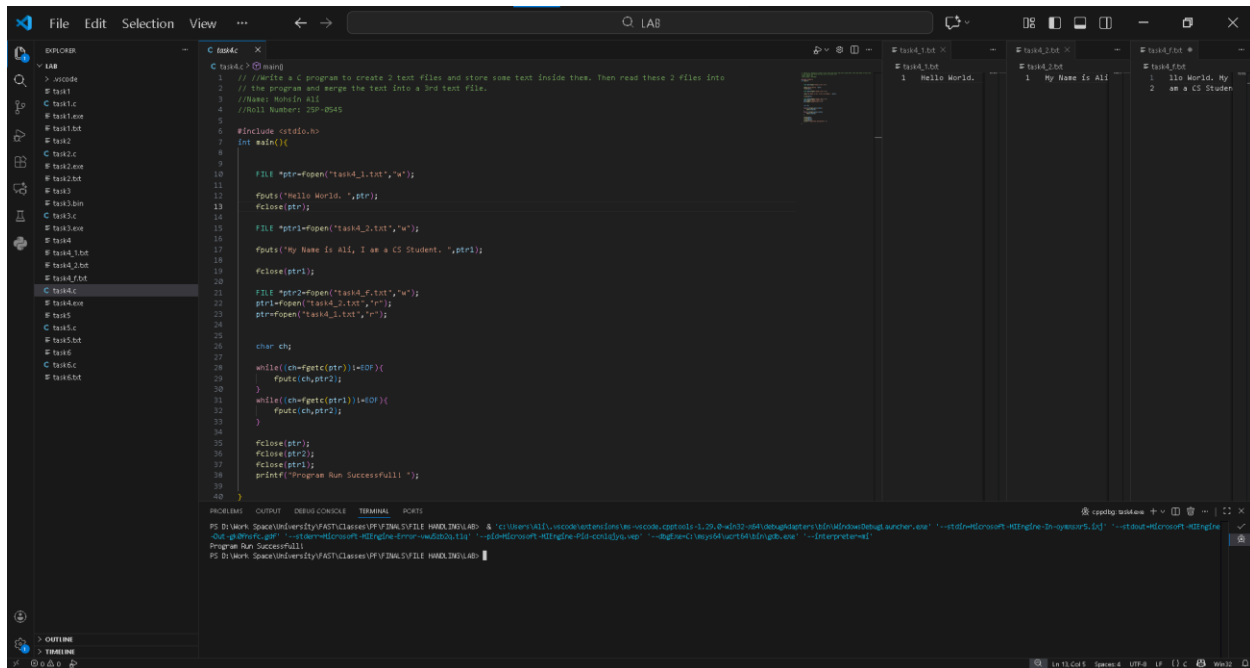
```
    ptr=fopen("task3.bin","rb");
```

```
    if(ptr==NULL){
```

```
        printf("Error Opening File: ");
```

```
}  
  
struct student s_read[3];  
  
fread(s_read,sizeof(struct student),3,ptr);  
  
//Display  
  
for(int i=0;i<3;i++){  
    printf("Name: %s : ",s_read[i].name);  
    printf("Marks : %d : \n",s_read[i].marks);  
  
}  
  
}
```


Q#4: Write a C program to create 2 text files and store some text inside them. Then read these 2 files into the program and merge the text into a 3rd text file.



The screenshot shows a Visual Studio Code editor with a C program in the main editor and three text files in the Explorer sidebar. The C program, named 'task4.c', is a C program that creates two text files, 'task4_1.txt' and 'task4_2.txt', and then reads their contents into a third file, 'task4_3.txt'. The program uses the following code:

```
1 // //Write a C program to create 2 text files and store some text inside them. Then read these 2 files into
2 // the program and merge the text into a 3rd text file.
3 //Name: Mohsin Ali
4 //Roll Number: 25P-0545
5
6 #include <stdio.h>
7
8 int main(){
9
10     FILE *ptr=fopen("task4_1.txt","w");
11
12     fputs("Hello World. ",ptr);
13     fclose(ptr);
14
15     FILE *ptr1=fopen("task4_2.txt","w");
16
17     fputs("My Name is Ali, I am a CS Student. ",ptr1);
18     fclose(ptr1);
19
20     FILE *ptr2=fopen("task4_3.txt","w");
21     ptr=fopen("task4_1.txt","r");
22     ptr=fopen("task4_2.txt","r");
23
24     char ch;
25
26     while((ch=fgetc(ptr))!=EOF){
27         fputc(ch,ptr2);
28     }
29
30     while((ch=fgetc(ptr1))!=EOF){
31         fputc(ch,ptr2);
32     }
33
34     fclose(ptr);
35     fclose(ptr1);
36     fclose(ptr2);
37     printf("Program Run Successfully\n");
38 }
```

The Explorer sidebar shows three text files: 'task4_1.txt' containing 'Hello World.', 'task4_2.txt' containing 'My Name is Ali', and 'task4_3.txt' containing 'Hello World, My' and 'I am a CS Student'.

CODE:

```
// //Write a C program to create 2 text files and store some text inside them. Then read these 2
files into
```

```
// the program and merge the text into a 3rd text file.
```

```
//Name: Mohsin Ali
```

```
//Roll Number: 25P-0545
```

```
#include <stdio.h>
```

```
int main(){
```

```
    FILE *ptr=fopen("task4_1.txt","w");
```

```
    fputs("Hello World. ",ptr);
```

```
    fclose(ptr);
```

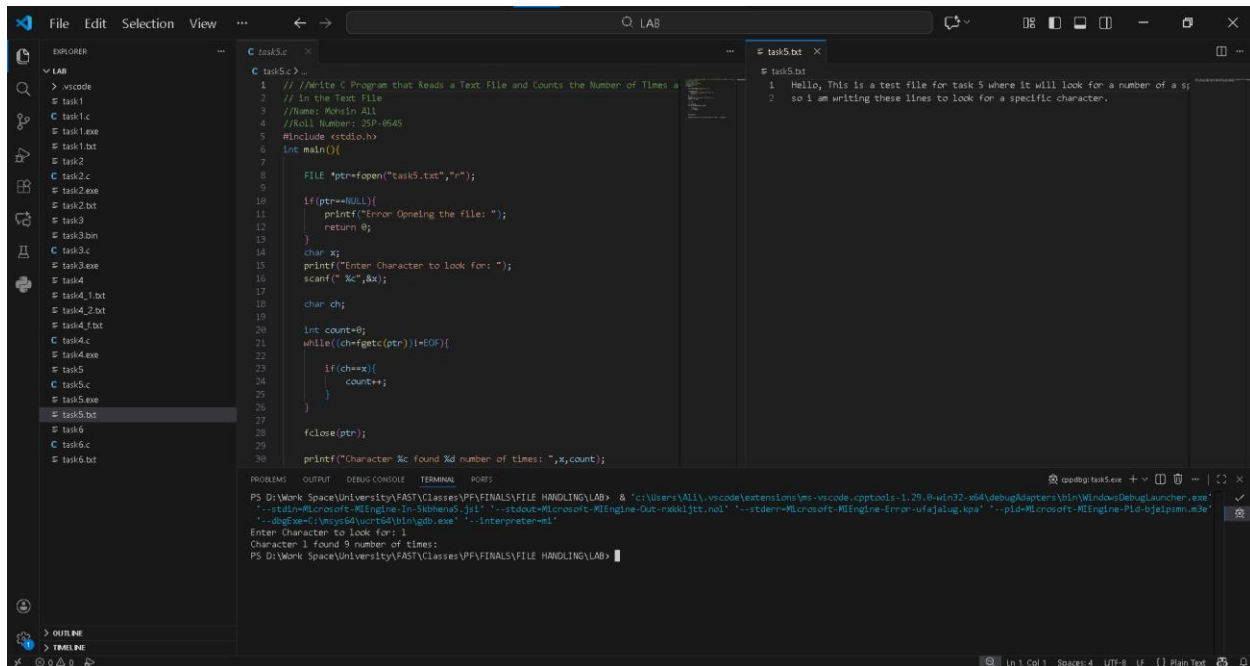
```
    FILE *ptr1=fopen("task4_2.txt","w");
```

```
    fputs("My Name is Ali, I am a CS Student. ",ptr1);
```

```
    fclose(ptr1);
```

```
FILE *ptr2=fopen("task4_f.txt","w");  
ptr1=fopen("task4_2.txt","r");  
ptr=fopen("task4_1.txt","r");  
char ch;  
while((ch=fgetc(ptr))!=EOF){  
    fputc(ch,ptr2);  
}  
while((ch=fgetc(ptr1))!=EOF){  
    fputc(ch,ptr2);  
}  
fclose(ptr);  
fclose(ptr2);  
fclose(ptr1);  
printf("Program Run Successfull! ");  
}
```

Q#5: Write C Program that Reads a Text File and Counts the Number of Times a Certain Letter Appears in the Text File



The screenshot shows a Visual Studio Code editor with two files open: `task5.c` and `task5.txt`. The `task5.c` file contains a C program that reads a text file and counts the occurrences of a specific character. The `task5.txt` file contains the text: "Hello, This is a test file for task 5 where it will look for a number of a s; so I am writing these lines to look for a specific character." The terminal output shows the program's execution, including the prompt "Enter Character to look for: l" and the result "Character l found 9 number of times: 9".

```
1 // //Write C Program that Reads a Text File and Counts the Number of Times a
2 // in the Text File
3 //Name: Mohsin Ali
4 //Roll Number: 25P-0545
5 #include <stdio.h>
6 int main()
7 {
8     FILE *ptr=fopen("task5.txt","r");
9
10    if(ptr==NULL){
11        printf("Error Opneing the file: ");
12        return 0;
13    }
14    char x;
15    printf("Enter Character to look for: ");
16    scanf("%c",&x);
17
18    char ch;
19
20    int count=0;
21    while((ch=fgetc(ptr))!=EOF){
22
23        if(ch==x){
24            count++;
25        }
26    }
27    fclose(ptr);
28    printf("Character %c found %d number of times: %d\n",x,count);
29 }
```

```
PS D:\Work Space\University\FAST\Classes\PP\FINAL\FILE HANDLING\LAB> & 'c:\Users\ALI\vscode\extensions\ms-vscode.cpptools-1.29.0-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe'
'--stdin=Microsoft-MIEngine-In-Skibana5-jul' '--stdout=Microsoft-MIEngine-Out-maki1jtt.nod' '--stderr=Microsoft-MIEngine-Error-ufajalug.kpa' '--pid=Microsoft-MIEngine-Pid-bjeipsm.mde'
'--debugName=C:\ProgramData\Microsoft\Windows\WinSxS\x-ww7-q932-964\debugAdapters\bin\WindowsDebugLauncher.exe' '--interpreter=ml'
Enter Character to look for: l
Character l found 9 number of times:
PS D:\Work Space\University\FAST\Classes\PP\FINAL\FILE HANDLING\LAB>
```

CODE:

// //Write C Program that Reads a Text File and Counts the Number of Times a Certain Letter Appears

// in the Text File

//Name: Mohsin Ali

//Roll Number: 25P-0545

#include <stdio.h>

int main(){

FILE *ptr=fopen("task5.txt","r");

if(ptr==NULL){

printf("Error Opneing the file: ");

return 0;

}

char x;

```
printf("Enter Character to look for: ");  
scanf(" %c",&x);  
char ch;  
int count=0;  
while((ch=fgetc(ptr))!=EOF){  
    if(ch==x){  
        count++;  
    }  
}  
fclose(ptr);  
printf("Character %c found %d number of times: ",x,count);  
}
```

Q#6: Consider a String entered by the user which he wants to encrypt and then decrypt the information. Perform this functionality via the filling technique by first inserting the encrypted text in the file then read the encrypted text to decode it back to string.

```
C task6.c
1 // Consider a String entered by the user which he wants to encrypt and then
2 // then read the encrypted text to decode it back to string.
3
4 //Name: Mohsin Ali
5 //Roll Number: 25P-0545
6 #include <stdio.h>
7 #include <string.h>
8 int main()
9 {
10     char word[256];
11
12     printf("Enter Word: ");
13
14     fgets(word,256,stdin);
15
16     word[strlen(word, "\n")] = '\0';
17
18     int len=strlen(word);
19
20     for(int i=0;i<len;i++){
21         word[i]=word[i]^'A';
22     }
23     printf("Encrypted text: \n");
24     for(int i=0;i<len;i++){
25         printf("%c",word[i]);
26     }
27
28     FILE *ptr=fopen("task6.txt","w");
29     fputs(word,ptr);
30 }
```

```
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB> & 'c:\Users\Ali\vscode\extensions\ms-vscode.cpptools-1.29.8-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe'
'--stdout=Microsoft.VisualStudio.Debugger.Out-djgump.wd' '--stderr=Microsoft.VisualStudio.Debugger.Out-djgump.wd' '--pid=Microsoft.VisualStudio.Debugger.Launcher.exe'
Enter Word: ABC
Encrypted text:
BDF
Decrypted Text:
ABC
PS D:\Work Space\University\FAST\Classes\PF\FINALS\FILE HANDLING\LAB>
```

CODE:

```
// Consider a String entered by the user which he wants to encrypt and then decrypt the information.
```

```
// Perform this functionality via the filling technique by first inserting the encrypted text in the file
```

```
// then read the encrypted text to decode it back to string.
```

```
//Name: Mohsin Ali
```

```
//Roll Number: 25P-0545
```

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main(){
```

```
    char word[256];
```

```
    printf("Enter Word: ");
```

```
fgets(word,256,stdin);
```

```
word[strcspn(word, "\n")] = '\0';
```

```
int ln=strlen(word);
```

```
for(int i=0;word[i]!='\0';i++){
```

```
    word[i]+=i+1;
```

```
}
```

```
printf("\nEncrypted text: \n");
```

```
for(int i=0;i<ln;i++){
```

```
    printf("%c",word[i]);
```

```
}
```

```
FILE *ptr=fopen("task6.txt","w");
```

```
fputs(word,ptr);
```

```
fclose(ptr);
```

```
ptr=fopen("task6.txt","r");
```

```
for(int i=0;word[i]!='\0';i++){
```

```
    word[i]-=(i+1);
```

```
}  
printf("\nDecrypted Text: \n");  
  
printf("%s",word);  
}
```