```
/* FILE (text or binary)*/

// input: io.txt
// output:  out.txt
// a console: con (nick name)



// Case 1. user -> con  => con  (no files are used here) [will not b considered]
// Case 2. user -> con => out.txt
// Case 3. io.txt => out.txt    // file copy
// Case 4. io.txt => con

//Case 4.1:
#include <stdio.h>
int main()
{
   FILE *fp;    // FILE is a system defined data type. Only for files systems.
   char ch;
   fp=fopen("ram.txt","r");     //ram.txt   "hi ayanEOF"
   // "r" reading "w" writing "a" appending

   while(1)
   {
        ch=fgetc(fp);
        if(ch==EOF) break;
        printf("%c",ch);
   }
   fclose(fp);
   return 0;
```

```
}

//Case 4.2:  WAP that will count the no of characters, spaces, new lines, tabs, ',' an so on..
#include <stdio.h>
int main()
{
   FILE *fp;    // FILE is a system defined data type. Only for files systems.
   char ch;
   int nos=0, nota=0, nol=0, noc=0, nod=0;

   fp=fopen("ram.txt","r");     //ram.txt   "hi, .aya.na dey  EOF"
   // "r" reading "w" writing "a" appending

   while(1)
   {
        ch=fgetc(fp);
        if(ch==EOF) break;
        //printf("%c",ch);
        switch(ch)
        {
           case ' ': nos++; break;
           case '\t': notab++; break;
           case '\n': nol++; break;
           case ',': noc++; break;
           case '.': nod++; break;
           default:
        }
   }
   fclose(fp);
```

```
   printf("No of tabs are %d", nos);
   printf("No of tabs are %d", notab);
   printf("No of tabs are %d", nol);
   printf("No of tabs are %d", noc);
   printf("No of tabs are %d", nod);
   return 0;
}

// Case 3. io.txt => out.txt    // file copy
#include <stdio.h>
int main()
{
   FILE *fp;    // FILE is a system defined data type. Only for files systems.
   char ch;
   fp=fopen("io.txt","r");     //io.txt   "hi ayanEOF"
   if(fp==NULL)
   {
      printf("ERROR: SOURCE FILE NOT FOUND\n");
      exit(1);
   }

   ft=fopen("out.txt","w");
   if(ft==NULL)
   {
      printf("ERROR: TARGET FILE NOT FOUND\n");
      fclose(fp);
      exit(1);
   }
```

```
   while(1)
   {
        ch=fgetc(fp);
        if(ch==EOF) break;
        else
         fputc(ch, ft);
   }
   fclose(fp);
   fclose(ft);
   return 0;
}

// Case 2.1. user -> con => out.txt         // "Hi how are you" => out.txt
#include <stdio.h>
#include <string.h>
int main()
{
   FILE *fp;    // FILE is a system defined data type. Only for files systems.
   char s[100];

   fp=fopen("ram.txt","w");
   if(fp==NULL)
   {
      printf("ERROR: FILE NOT FOUND\n");
      exit(1);
   }

   printf("\nPlease Enter some texts\n");
```

```
   while(strlen(gets(s))>0)
   {
        fputs(s,fp);
        fputs("\n",fp);
   }
   fclose(fp);
   return 0;
}

// hi how are you
// r u from kolkata? yes or no ? ...
//



// Case 2.2. user -> con => out.txt          // out.txt => console
#include <stdio.h>
#include <string.h>
int main()
{
   FILE *fp;    // FILE is a system defined data type. Only for files systems.
   char s[100];

   fp=fopen("ram.txt","r");
   if(fp==NULL)
   {
      printf("ERROR: FILE NOT FOUND\n");
      exit(1);
   }
```

```
   while(fgets(s,99,fp))!=NULL)
     puts(s);

   fclose(fp);
   return 0;
}



// Binary file and recursion

//Break will exit the control from the loop or switch
//Return will exit the control from any function to Main
//Exit will return the control from any location to compiler
```