

190905104

CD Lab 2

Sample program

Program to remove single and multiline comments from a given 'C' file.

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

int main(){
    FILE *fa,*fb;
    int ca,cb;
    char* fname[100];
    printf("Enter file name\n");
    scanf("%s", fname);
    fa = fopen(fname,"r");
    if(fa == NULL){
        printf("Cannot open file \n");
        exit(0);
    }
    fb = fopen("sampleout.c","w");
    ca = getc(fa);
    while(ca != EOF){
        if(ca == '"'){
            putc(ca,fb);
            while(ca == '"') ca = getc(fa);
        }
        if(ca == '/'){
            cb = getc(fa);
            if(cb == '/'){
                while(ca != '\n')
                    ca = getc(fa);
            }
            else if(cb == '*'){
                do{
                    while(ca != '*')
                        ca = getc(fa);
                    ca = getc(fa);
                }while(ca != '/');
            }
            else{
                putc(ca,fb);
                putc(cb,fb);
            }
        }
        else putc(ca,fb);
        ca = getc(fa);
    }
    fclose(fa);
    fclose(fb);
}
```

```

    return 0;
}

```

```

C samplein.c x
C samplein.c > main()
1 // This is a single line comment
2 /* *****This is a
3 *****Multiline Comment
4 **** */
5 #include <stdio.h>
6 void main()
7 {
8 FILE *fopen(), *fp;
9 int c ;
10 fp = fopen( "prog.c", "r" ); //Comment
11
12 13
13 c = getc( fp ) ;
14 while ( c != EOF )
15 {
16     putchar( c );
17     c = getc ( fp );
18 } /*multiline
19 comment */
20 fclose( fp );
21
22
C sampleout.c 5 x
C sampleout.c > ...
1
2 #include <stdio.h>
3 void main()
4 {
5 FILE *fopen(), *fp;
6 int c ;
7 fp = fopen( "prog.c", "r" );
8 13
9
10 c = getc( fp ) ;
11 while ( c != EOF )
12 {
13     putchar( c );
14     c = getc ( fp );
15 }
16 fclose( fp );
17

```

1) That takes a file as input and replaces blank spaces and tabs by single space and writes the output to a file.

```

#include <stdio.h>
#include <stdlib.h>

int main(){
    FILE *fa,*fb;
    int ca, cb;
    char* fname[100];
    printf("Enter file name\n");
    scanf("%s", fname);
    fa = fopen(fname,"r");
    if(fa == NULL){
        printf("Cannot open file \n");
        exit(0);
    }
    fb = fopen("q1out.c", "w");

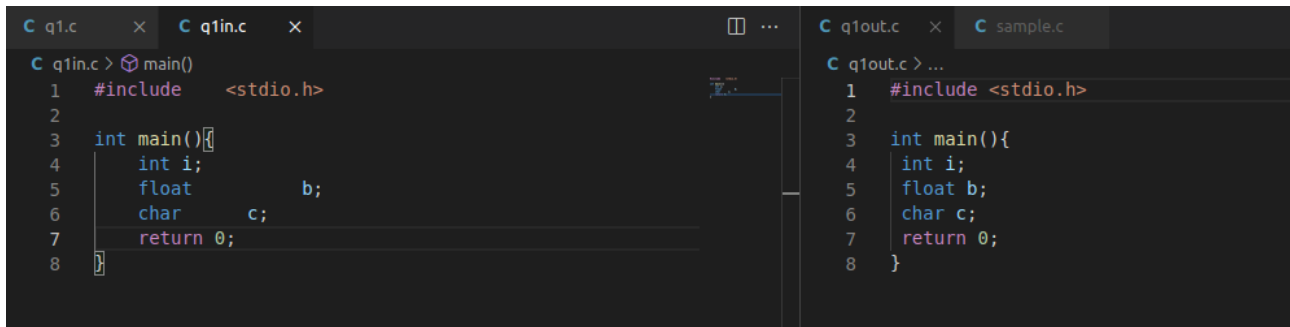
    ca = getc(fa);
    while(ca != EOF){
        if(ca == ' '){
            putc(' ', fb);
            while(ca == ' '){
                ca = getc(fa);
            }
            putc(ca, fb);
        }
        else{
            putc(ca, fb);
        }
        ca = getc(fa);
    }
}

```

```

fclose(fa);
fclose(fb);
}

```



2) To discard preprocessor directives from the given input 'C' file.

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

```

```

const char *directives[] = {"#define", "#include"};

```

```

int isDirective(const char *str){
    for(int i = 0; i < sizeof(directives)/sizeof(char*); i++){
        if(strncmp(str, directives[i], strlen(directives[i])) == 0){
            return 1;
        }
    }
    return 0;
}

```

```

int main(){
    FILE *fa, *fb;
    char buff[2048];
    char filename[100];
    printf("enter filename to open:\n");
    scanf("%s", filename);
    fa = fopen(filename, "r");

    fb = fopen("q2out.c", "w");
    if(!fa || !fb){
        printf("cannot open file\n");
        exit(0);
    }
    while(fgets(buff, 2048, fa)){
        if(!isDirective(buff))
            fputs(buff, fb);
    }
    fclose(fa);
    fclose(fb);
}

```

```

C q1in.c > ...
1  #include <stdio.h>
2  #define size 100
3  #inl <stdlib.h>
4
5  int main(){
6      int i;
7      float      b;
8      char      c;
9      return 0;
10 }

C q2out.c > ...
1  #inl <stdlib.h>
2
3  int main(){
4      int i;
5      float      b;
6      char      c;
7      return 0;
8  }

```

3) That takes C program as input, recognizes all the keywords and prints them in upper case.

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>

```

```

const char *keywords[] = {"printf", "int", "float", "return", "break", "continue", "if", "else",
"for", "while"};

```

```

int isKeyword(const char* str){
    for(int i = 0;i<sizeof(keywords)/sizeof(char*);i++){
        if(strncmp(str,keywords[i],strlen(keywords[i])) == 0)
            return 1;
    }
    return 0;
}

```

```

void stringToUpper(char *str,const int n){
    for(int i = 0;i<n;i++){
        str[i] = toupper(str[i]);
    }
}

```

```

int main(){
    char buff[2048];
    FILE *fa;
    char fname[100],c;
    printf("Enter file name\n");
    scanf("%s",fname);
    fa = fopen(fname,"r");
    if(!fa){
        printf("cannot open file\n");
        exit(0);
    }

    int line =1,col=1,j=0;
    printf("Keywords are:\n");

    while((c = fgetc(fa)) != EOF){
        if(isalpha(c)){
            buff[j++] = c;
        }
        else{

```

```

        buff[j] = '\0';
        if(isKeyword(buff)){
            stringToUpper(buff,j);
            printf("%s: at(line %d, col %d)\n",buff,line,col-j);
        }
        j = 0;
    }
    if(c == '\n'){
        line++;
        col = 1;
    }
    else col++;
}
fclose(fa);
return 0;
}

```

```

ugcse@prg28:~/190905104_CD/Lab2$ gcc q3.c -o q3
ugcse@prg28:~/190905104_CD/Lab2$ ./q3
Enter file name
q3in.c
Keywords are:
INT: at(line 3, col 1)
INT: at(line 4, col 5)
FOR: at(line 5, col 5)
INT: at(line 5, col 9)
IF: at(line 6, col 9)
PRINTF: at(line 7, col 13)
ELSE: at(line 9, col 9)
CONTINUE: at(line 10, col 13)
RETURN: at(line 13, col 5)
ugcse@prg28:~/190905104_CD/Lab2$

```