Session 1: Scanning and Filtering a Source Program

I. OBJECTIVES

To develop a program which can filter comments and white space characters from a source program

II. DEMONSTRATION OF USEFUL RESOURCES

Extracting the sequence of occurrences of a specified character from a source program

Sample Input: datafile1.c

datafile1.c

```
#include <stdio.h>
int main(void)
{
   FILE *p1,*p2; char c;

   p1 = fopen("datafile1.c", "r");
   p2 = fopen("parentheses.txt","w");

   if(!p1) printf("\nFile can't be opened!");
   else {
         while((c = fgetc(p1)) != EOF) {
                  if ((c == '(') || (c == ')'))
                  fputc(c, p2);  } }
   fclose(p1);
   fclose(p2);

   p2 = fopen("parentheses.txt","r");
   while((c=fgetc(p2))!=EOF)
                  printf("%c",c);
   fclose(p2);

   return 0;
}
```

Output of the program: ()()()()(())((()))()()()()(())()()

III. LAB EXERCISE

1. Write a program to print the header files used in a source program.

Sample Input: *input.c*

```
#include <stdio.h>
int main()
{
  // printf() displays the string inside quotation
printf("Hello, World!");
return 0;
}
```

Sample Output: *stdio.h*

2. Write a program to add line numbers to a source program.

Sample Input: *input.c*

Sample Output:

```
1: #include <stdio.h>
2: int main()
3: {
4:  // printf() displays the string inside quotation
5: printf("Hello, World!");
6:    return 0;
7: }
```

IV. ASSIGNMENT #1:

A C source program with single and multiple line comments is given. As the first step toward compilation you need to remove the comments and white space (extra spaces, tabs and newline characters). Develop a program that takes as input file the given source program and produces a filtered file as stated above. The program must also display both the files.

Sample Input: *input1.c*

```
#include<stdio.h>

int    main(void)
{

// Single    Line Comment

printf   ("Hello");
/* Multi
     Line
                    Comment
*/
printf("World");
return 0;
}
```

Sample Output: *output.txt*

```
#include<stdio.h> int main(void) { printf ("Hello"); printf("World"); return 0; }
```