

Name : Vivek Singh
RegNumber : 230905408
Roll Number: A50

Lab 2: PRELIMINARY SCANNING APPLICATIONS

Q1)Write a program that takes a file as input and replaces blank spaces and tabs by single space and writes the output to a file.

Code:

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main(int argc, char *argv[]){
5      FILE *fr, *fw;
6      char ca, cb;
7
8      // fr = fopen(argv[1], 'r');
9      // fw = fopen(argv[2], 'w');
10
11     fr = fopen("q1.c","r");
12     fw = fopen("rq1.c","w");
13
14     ca = getc(fr);
15     while(ca!=EOF){
16         if(ca==' '){
17             if((cb= getc(fr))==' '){
18                 while((cb = getc(fr)) == ' ');
19                 putc('\t', fw);
20             }
21             putc(cb, fw);
22         } else {
23             putc(ca, fw);
24         }
25         ca = getc(fr);
26     }
27
28     fclose(fr);
29     fclose(fw);
30     return 0;
31 }
```

Output:

```
1  #include<stdio.h>
2  #include<string.h>
3
4  int main(int argc, char* argv[]){
5      FILE* fr, *fw;
6      char ca, cb;
7
8      //fr=fopen(argv[1], 'r');
9      //fw=fopen(argv[2], 'w');
10
11     fr=fopen("q1.c", "r");
12     fw=fopen("rq1.c", "w");
13
14     ca=getc(fr);
15     while(ca!=EOF){
16         if(ca=='\n'){
17             if((cb=getc(fr))=='\n'){
18                 while((cb=getc(fr))=='\n');
19                 putc('\t', fw);
20             }
21             putc(cb, fw);
22         }else{
23             putc(ca, fw);
24         }
25         ca=getc(fr);
26     }
27
28     fclose(fr);
29     fclose(fw);
30     return 0;
31 }
```

Q2) Write a program To discard preprocessor directives from the given input 'C' file.

Code:

```
C q2.c > main()
1  #include <stdio.h>
2  #include <ctype.h>
3
4  int main() {
5      FILE *fr, *fw;
6      char line[1024];
7
8      fr = fopen("q1.c", "r");
9      fw = fopen("qr2.c", "w");
10
11     while (fgets(line, sizeof(line), fr) != NULL) {
12         char *ptr = line;
13
14         while (*ptr && isspace(*ptr)) {
15             ptr++;
16         }
17
18         if (*ptr != '#') {
19             fputs(line, fw);
20         }
21     }
22
23     fclose(fr);
24     fclose(fw);
25
26     return 0;
27 }
```

Output :

```
1
2 int main(int argc, char *argv[]){
3     FILE *fr, *fw;
4     char ca, cb;
5
6     // fr = fopen(argv[1], 'r');
7     // fw = fopen(argv[2], 'w');
8
9     fr = fopen("q1.c", "r");
10    fw = fopen("rq1.c", "w");
11
12    ca = getc(fr);
13    while(ca != EOF){
14        if(ca == ' '){
15            if((cb = getc(fr)) == ' '){
16                while((cb = getc(fr)) == ' ');
17                putc('\t', fw);
18            }
19            putc(cb, fw);
20        } else {
21            putc(ca, fw);
22        }
23        ca = getc(fr);
24    }
25
26    fclose(fr);
27    fclose(fw);
28    return 0;
29 }
```

Q3) Write a program That takes C program as input, recognizes all the keywords and prints them in upper case.

Code:

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <ctype.h>
4
5  int isKeyword(char *w) {
6  >   char *k[] = { ...
13
14   int l = 0, r = 31, m, c;
15
16   while (l <= r) {
17       m = l + (r - l) / 2;
18       c = strcmp(w, k[m]);
19       if (c == 0) return 1;
20       if (c < 0) r = m - 1;
21       else l = m + 1;
22   }
23   return 0;
24 }
25
26 int main(int argc, char *argv[]) {
27     FILE *f = fopen(argv[1], "r");
28     char ch, w[100];
29     int i = 0, j;
30
31     while ((ch = fgetc(f)) != EOF) {
32         if (isalnum(ch) || ch == '_') {
33             w[i++] = ch;
34         } else {
35             if (i > 0) {
36                 w[i] = '\0';
37                 if (isKeyword(w)) {
38                     for (j = 0; w[j]; j++)
39                         w[j] = toupper(w[j]);
40                 }
41                 printf("%s", w);
42                 i = 0;
43             }
44             printf("%c", ch);
45         }
46     }
47
48     fclose(f);
49     return 0;
50 }
```

Output :

```
● CD_A2@CL3-15:~/Documents/230905408/lab2$ ./q3 q1.c
#include <stdio.h>
#include <string.h>

INT main(INT argc, CHAR *argv[]){
    FILE *fr, *fw;
    CHAR ca, cb;

    // fr = fopen(argv[1], 'r');
    // fw = fopen(argv[2], 'w');

    fr = fopen("q1.c", "r");
    fw = fopen("rq1.c", "w");

    ca = getc(fr);
    WHILE(ca!=EOF){
        IF(ca==' '){
            IF((cb= getc(fr))== ' '){
                WHILE((cb = getc(fr)) == ' ');
                putc('\t', fw);
            }
            putc(cb, fw);
        } ELSE {
            putc(ca, fw);
        }
        ca = getc(fr);
    }

    fclose(fr);
    fclose(fw);
    RETURN 0;
}
○ } CD_A2@CL3-15:~/Documents/230905408/lab2$
```