

Question 1

// Program 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(){
    // declaring file pointers and variables
    FILE *fa, *fa_, *fb;
    int ca, ca_;

    // opening files
    fa = fopen("q1_src.c", "r");
    fa_ = fopen("q1_src.c", "r");
    fb = fopen("q1_res.c", "w+");

    // checking if file oprns
    if (fa == NULL){
        printf("Cannot open file \n");
        exit(0);
    }

    // going to first position of the file
    ca = getc(fa);

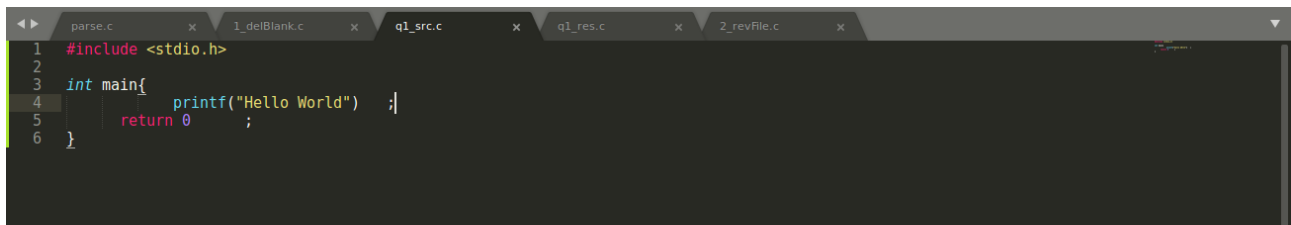
    // going to the second position of the file
    ca_ = getc(fa_);
    ca_ = getc(fa_);

    // going through the document
    while(ca != EOF){
        if(ca == ' ' && ca_ == ' '){ // if more than 2 consecutive blanks are found
            ca = getc(fa);
            ca_ = getc(fa_);
            continue;
        } else if (ca == '\t' && ca_ == '\t') { // if more than 2 consecutive tabs are found
            ca = getc(fa);
            ca_ = getc(fa_);
            continue;
        } else if (ca == '\t' && ca_ != '\t') { // condensing consecutive tabs
            putc(' ', fb);
            ca = getc(fa);
            ca_ = getc(fa_);
        } else { // writing all other characters into the file
            putc(ca, fb);
            ca = getc(fa);
            ca_ = getc(fa_);
        }
    }
}
```

```

        // closing files
        fclose(fa);
        fclose(fa_);
        fclose(fb);
        return 0;
}

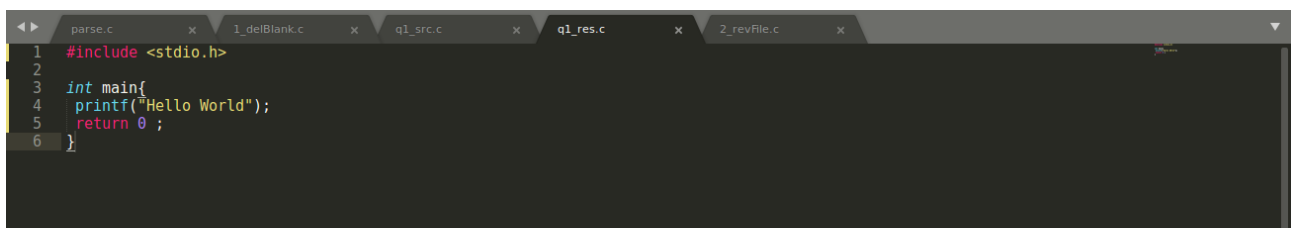
```



```

1 #include <stdio.h>
2
3 int main{
4     printf("Hello World") ;|
5     return 0 ;
6 }

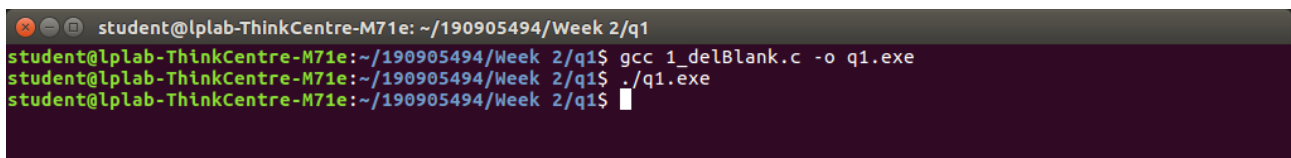
```



```

1 #include <stdio.h>
2
3 int main{
4     printf("Hello World");
5     return 0 ;
6 }

```



```

student@lplab-ThinkCentre-M71e: ~/190905494/Week 2/q1
student@lplab-ThinkCentre-M71e:~/190905494/Week 2/q1$ gcc 1_delBlank.c -o q1.exe
student@lplab-ThinkCentre-M71e:~/190905494/Week 2/q1$ ./q1.exe
student@lplab-ThinkCentre-M71e:~/190905494/Week 2/q1$

```

Question 2

// Program to discard preprocessor directives from the given input 'C' file.

```

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

int main(){
    // declaring file pointers and variables
    FILE *fa, *fa_, *fb;
    int ca, ca_, flag_1 = 0, flag_2 = 0;

    // opening files
    fa = fopen("q2_src.c", "r");
    fa_ = fopen("q2_src.c", "r");
    fb = fopen("q2_res.c", "w+");

    // checking if file oprns
    if (fa == NULL){
        printf("Cannot open file \n");
        exit(0);
    }
}

```

```

}

// initializing ca
ca = getc(fa);

// initializing ca_
ca_ = getc(fa_);

// going through the document
while(ca != EOF){

    // check if '#' is encountered
    if(ca == '#'){

        // check if it is actually a directive
        while(ca_ != '\n'){
            if(ca_ == '<') flag_1 = 1;
            if(ca_ == '>') flag_2 = 1;
            ca_ = getc(fa_);
        }

        ca_ = getc(fa_);

        if(flag_1==1 && flag_2==1){                // removing directive
            flag_2 = 0;
            flag_1 = 0;

            while(ca != '\n'){
                ca = getc(fa);
            }

            ca = getc(fa);
            continue;
        } else {                                // keeping sentence with #
            flag_2 = 0;
            flag_1 = 0;

            while(ca != '\n'){
                putc(ca, fb);
                ca = getc(fa);
            }

            putc(ca, fb);
            ca = getc(fa);
            continue;
        }
    }

    // keeping all other charaters
    putc(ca, fb);
    ca = getc(fa);
}

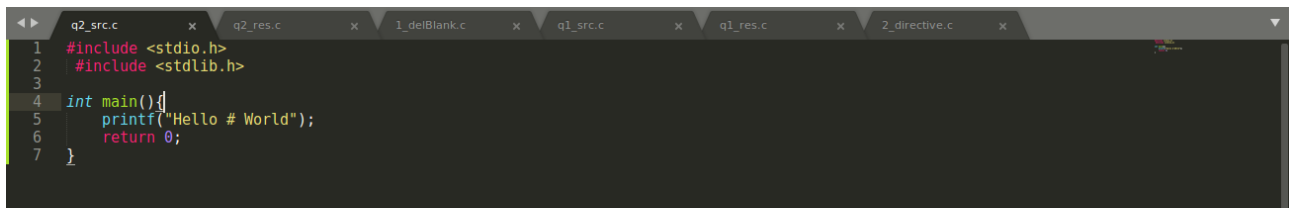
```

```

        ca_ = getc(fa_);
    }

    // closing files
    fclose(fa);
    fclose(fa_);
    fclose(fb);
    return 0;
}

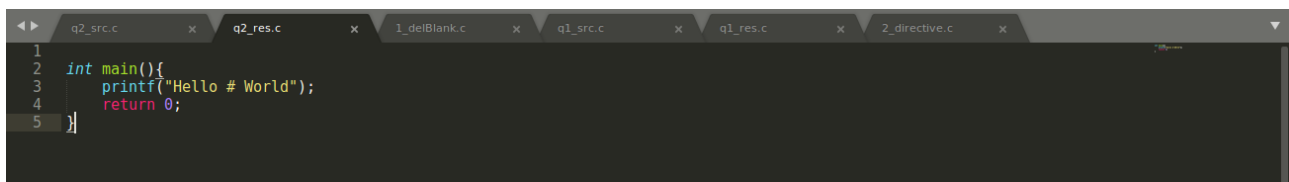
```



```

q2_src.c
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(){
5      printf("Hello # World");
6      return 0;
7  }

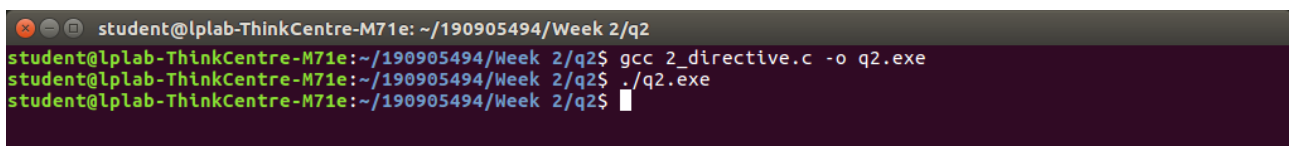
```



```

q2_src.c
1  int main(){
2      printf("Hello # World");
3      return 0;
4  }
5

```



```

student@lplab-ThinkCentre-M71e: ~/190905494/Week 2/q2
student@lplab-ThinkCentre-M71e:~/190905494/Week 2/q2$ gcc 2_directive.c -o q2.exe
student@lplab-ThinkCentre-M71e:~/190905494/Week 2/q2$ ./q2.exe
Hello # World
student@lplab-ThinkCentre-M71e:~/190905494/Week 2/q2$

```

Question 3

// Program 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>

int main(){

    // declaring file pointers and variables
    FILE *fa;
    int ca, i=0, j, k;
    char buffer[100], ch, keywords[10][100] = {"int", "main", "if", "while", "do", "FILE",
"char", "float", "continue", "break"};

    // opening files
    fa = fopen("q3.c", "r");

```

```

// checking if file oprns
if (fa == NULL){
    printf("Cannot open file \n");
    exit(0);
}

// initializing ca
ca = getc(fa);

printf("=== Keyword Found in the Document ===\n");

// going through the document
while(ca != EOF){

    // skipping non alphanumeric
    if(isalpha(ca) == 0) {
        i=0;
        ca = getc(fa);
        continue;
    }

    // creating word
    while(isalpha(ca) != 0){
        buffer[i++] = ca;
        ca = getc(fa);
    }

    // adding null char to end
    buffer[i] = '\0';

    // matching and printing buffer
    for(j = 0; j<10; j++){
        if(strcmp(buffer, keywords[j]) == 0){
            for(k = 0; buffer[k] != '\0'; k++){
                printf("%c", toupper(ch));
            }
            printf("\n");
        }
    }

}

// closing files
fclose(fa);
return 0;
}

```

```
q2.c x 1_delBlank.c x q1_src.c x q1_res.c x 2_directive.c x q2_src.c x q2_res.c x 3_keywords.c x
1 int main(){
2
3 }
```

```
student@lplab-ThinkCentre-M71e: ~/190905494/Week 2
student@lplab-ThinkCentre-M71e:~/190905494/Week 2$ ./q3.exe
=== Keyword Found in the Document ===
INT
MAIN
student@lplab-ThinkCentre-M71e:~/190905494/Week 2$
```