Name:JACOB V SANOJ	SRN:PES1UG20EC083	Section:F1
	Date:10-07-2021	Week Number:9

1 Write a C program to merge contents of two files into a third file.

Hint: Create three files- 2 files(file1.txt and file2.txt) with contents and third file(file3.txt) to merge contents of other two files(file1.txt and file2.txt).

Input:

gedit file1.txt

Hi,Good morning!

Have a nice day

gedit file2.txt

Welcome to C programming- file handling concepts

gedit file3.txt

//empty file

Enter the 1st file name : file1.txt

Enter the 2nd file name : file2.txt

Enter the new file name to merge the two files:file3.txt

Output:

The two files merged into file3.txt file successfully..!!

//Third file-Merged contents of two files(file1.txt and file2.txt)

gedit file3.txt

Hi,Good morning!

Have a nice day

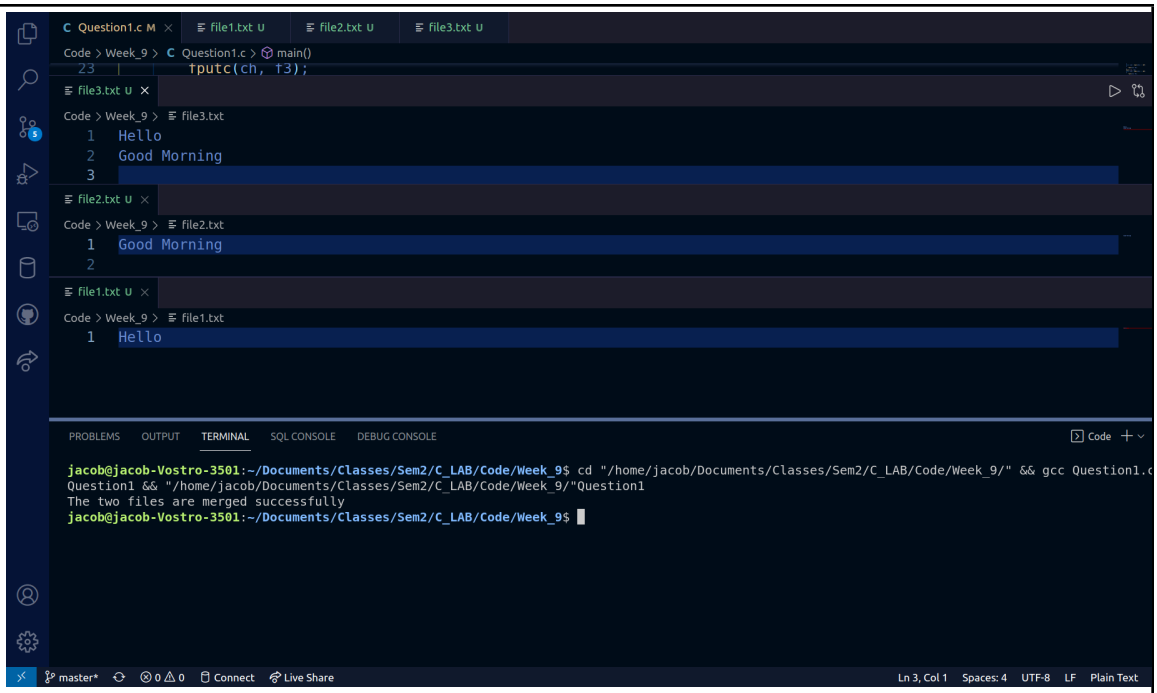
Welcome to C programming- file handling concepts

Program:

```c
#include <stdio.h>
int main()
{
```

```c
    FILE *f1, *f2, *f3;

    char ch, file1[20] = "file1.txt", file2[20] = "file2.txt",
file3[20] = "file3.txt";
    f1 = fopen(file1, "r");
    f2 = fopen(file2, "r");
    f3 = fopen(file3, "w");

    if (f1 == NULL || f2 == NULL || f3 == NULL)
    {
        printf("File does not exist!\n");
        return 0;
    }
    while ((ch = fgetc(f1)) != EOF)
    {
        fputc(ch, f3);
    }
    fputc('\n', f3);
    while ((ch = fgetc(f2)) != EOF)
    {
        fputc(ch, f3);
    }
    printf("The two files are merged successfully\n");
    fclose(f1);
    fclose(f2);
    fclose(f3);
}
```

Output Screenshot:



2 Write a C program to write multiple lines in a text file.

Input:

enter the filename

file.txt

Enter the number of lines to be written : 2

The lines are

hi hello

how are you

Output:

The content of the file file.txt is :

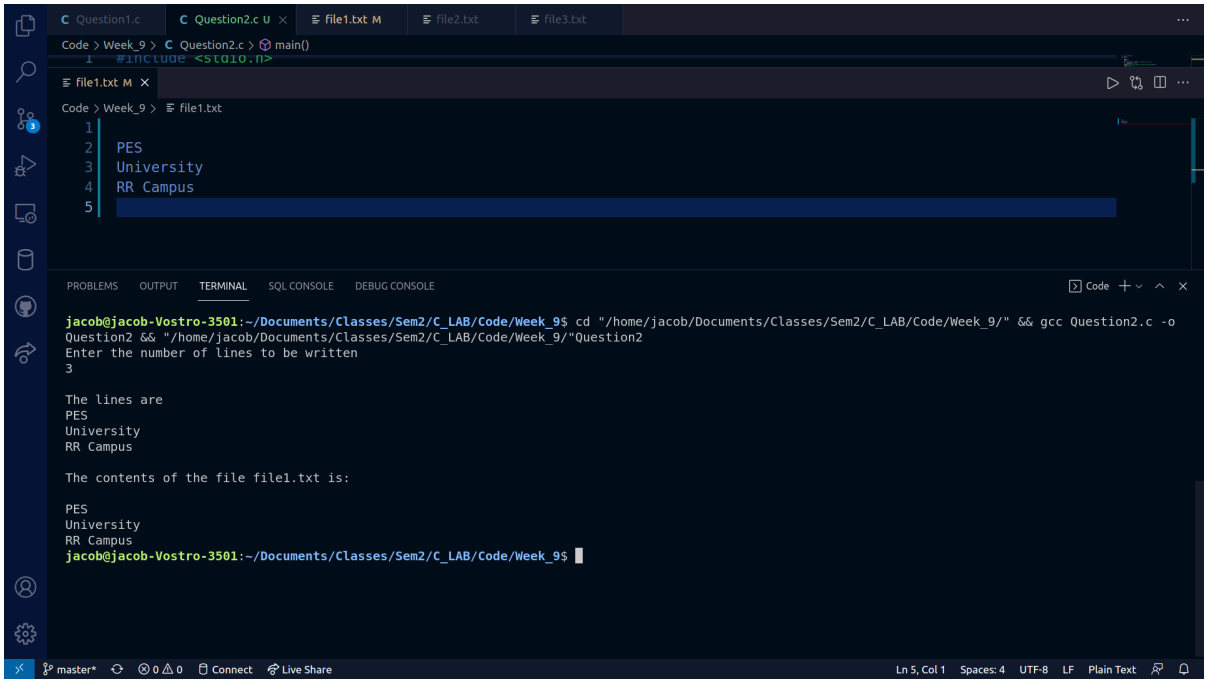
hi hello

how are you

Program:

```c
#include <stdio.h>
int main()
{
   FILE *p;
   int n;
   char str[100], name[20] = "file1.txt", str1;
   p = fopen(name, "w");
   printf("Enter the number of lines to be written\n");
   scanf("%d", &n);
   printf("\nThe lines are \n");
   for (int i = 0; i < n + 1; i++)
   {
       fgets(str, sizeof(str), stdin);
       fputs(str, p);
   }
   fclose(p);
   p = fopen(name, "r");
   printf("\nThe contents of the file %s is:\n", name);
   str1 = fgetc(p);
   while (str1 != EOF)
   {
       printf("%c", str1);
       str1 = fgetc(p);
   }
   fclose(p);
}
```

Output Screenshot:



3 Write a program to sort positive integers in the ascending order using insertion sort

Input:

Enter the number of elements u want to sort

5

Output:

Enter 5 elements

13

6

23

1

89

Before sorting

13

6

23

1

89

After sorting

1

6

13

23

89

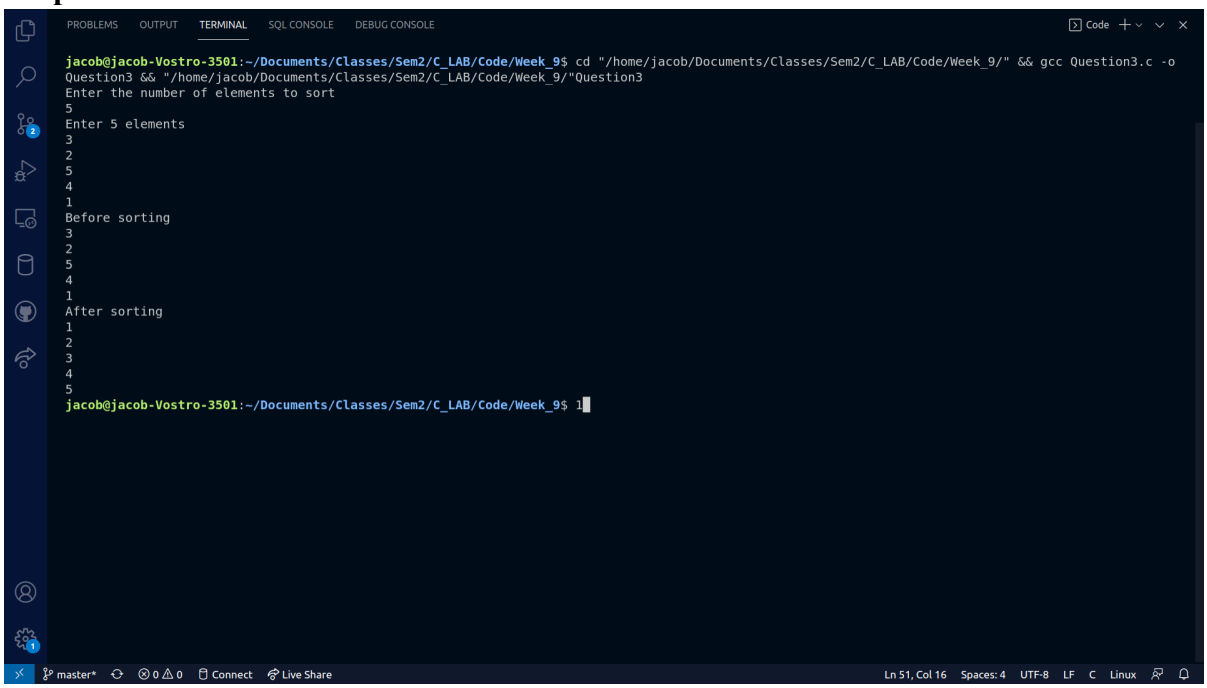
Program:

```
#include <stdio.h>

void read(int *a, int n)
{
   for (int i = 0; i < n; i++)
   {
       scanf("%d", &a[i]);
   }
}

void disp(int *a, int n)
```

```
{
    for (int i = 0; i < n; i++)
    {
        printf("%d\n", a[i]);
    }
}

void insertion_sort(int *a, int n)
{
    int j, temp;
    for (int i = 0; i < n; i++)
    {
        temp = a[i];
        j = i;
        while (j > 0 && temp < a[j - 1])
        {
            a[j] = a[j - 1];
            --j;
        }
        a[j] = temp;
    }
}

int main()
{
    int a[100], n;

    printf("Enter the number of elements to sort\n");
    scanf("%d", &n);

    printf("Enter %d elements\n", n);
    read(a, n);
```

```
    printf("Before sorting\n");
    disp(a, n);

    insertion_sort(a, n);

    printf("After sorting\n");
    disp(a, n);
}
```

Output Screenshot:



4 Write a bubblesort program to sort students details based on students roll number/name in the ascending order using array of pointers, by taking input from csv file and using callback to call two functions i)sort based on roll number ii) sort based on name.

Input:

stud.csv file

Output:

99 xx

7 bb

22 cc

45 zz

8 aa

12 ff

4 gg

3 dd

27 jj

1 kk

32 ee

Enter your option

1.sort on roll

2.sort on name

0.exit

1

```
1 kk

3 dd

4 gg

7 bb

8 aa

12 ff

22 cc

27 jj

32 ee

45 zz

99 xx

Enter your option
1.sort on roll
2.sort on name
0.exit
2
8 aa
```

7 bb

22 cc

3 dd

32 ee

12 ff

4 gg

27 jj

1 kk

99 xx

45 zz

Program:

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

typedef struct student
{
   int rno;
   char name[20];
} stud_t;
```

```c
void init_ptr(stud_t *s, stud_t *p[], int n)
{
    for (int i = 0; i < n; i++)
    {
        p[i] = &s[i];
    }
}
void disp(stud_t *p[], int n)
{
    for (int i = 0; i < n; i++)
    {
        printf("%d %s", p[i]->rno, p[i]->name);
    }
    printf("\n");
}
void bubble_sort_rno(stud_t *s[], int n)
{
    for (int i = 0; i < n - 1; i++)
    {
        for (int j = 0; j < n - 1; j++)
        {
            if (s[j]->rno > s[j + 1]->rno)
            {
                stud_t *temp = s[j];
                s[j] = s[j + 1];
                s[j + 1] = temp;
            }
        }
    }
}
void bubble_sort_name(stud_t *s[], int n)
{
```

```
    for (int i = 0; i < n - 1; i++)
    {
        for (int j = 0; j < n - 1; j++)
        {
            if (strcmp(s[j]->name, s[j + 1]->name) > 0)
            {
                stud_t *temp = s[j];
                s[j] = s[j + 1];
                s[j + 1] = temp;
            }
        }
    }
}
void bubblesort(void (*ptr)(), stud_t *s[], int n)
{
    (*ptr)(s, n);
}

int main()
{
    FILE *fp = fopen("student.csv", "r");
    char a[100];
    char *p;
    stud_t s[100];
    void (*ptr)() = &bubble_sort_rno;
    void (*ptr1)() = &bubble_sort_name;
    int i = 0;
    fgets(a, 100, fp);
    while (fgets(a, 100, fp))
    {
        p = strtok(a, ",");
        s[i].rno = atoi(p);
```

```
        p = strtok(NULL, ",");
        strcpy(s[i].name, p);
        ++i;
    }
    int n = i;
    fclose(fp);
    stud_t *ap[100];
    init_ptr(s, ap, n);
    disp(ap, n);
    int op;
    printf("Enter your option\n 1.sort on roll\n 2.sort on
name\n 0.exit\n");
    scanf("%d", &op);
    while (op)
    {
        switch (op)
        {
        case 1:
            bubblesort(ptr, ap, n);
            disp(ap, n);
            break;
        case 2:
            bubblesort(ptr1, ap, n);
            disp(ap, n);
            break;
        }
        printf("Enter your option\n 1.sort on roll\n 2.sort on
name\n 0.exit\n");
        scanf("%d", &op);
    }
    return 0;
}
```

Output Screenshot: