Name: SUNDEEP A	SRN: PES1UG20CS445	Section: O
	Date: 01-07-2021	Week Number: 9

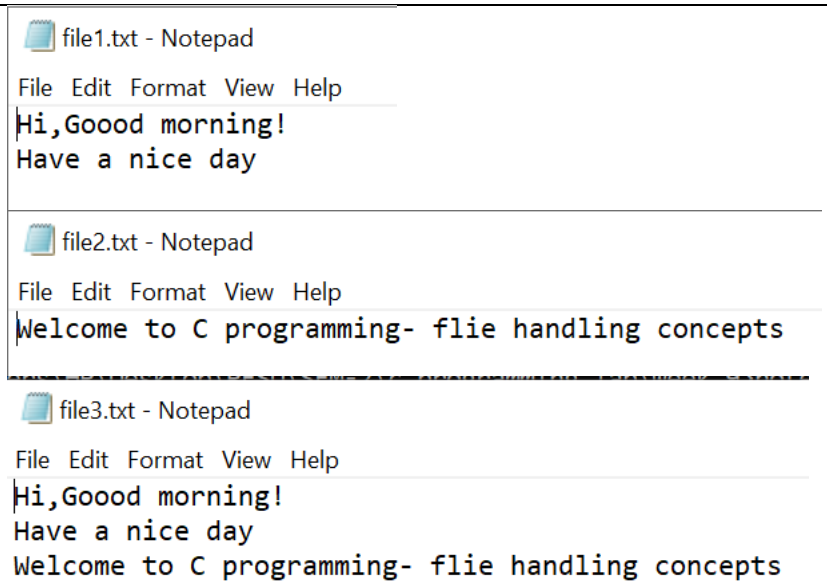
1	Write a C program to merge contents of two files into a third file.
	
	Hint: Create three files- 2 files(file1.txt and file2.txt) with contents and third file(file3.txt) to merge contents of other two files(file1.txt and file2.txt).
	
	Input:
	gedit file1.txt
	Hi,Good morning!
	Have a nice day
	gedit 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: **#include <stdio.h>** **#include <stdlib.h>** **void read(FILE *fp1,FILE *fp2,FILE *fp3)** **{ char c;** **while ((c = fgetc(fp1)) != EOF) //read content from file1**

```
 fputc(c, fp3);                          //append the contents of file1 to file3
 while ((c = fgetc(fp2)) != EOF)         //read content of file2
 fputc(c, fp3);                          //append the contents of file2 to file3
}
int main()
{
 char a[10],b[10],c[10];
 printf("Enter the 1st file name: ");          //storing file1 name in a
 scanf("%s",a);
 printf("Enter the 2nd file name: ");          //storing file2 name in b
 scanf("%s",b);
 printf("Enter the file name to merge contents of prev files: ");
 scanf("%s",c);
 FILE *fp1 = fopen(a, "r");
 FILE *fp2 = fopen(b, "r");
 FILE *fp3 = fopen(c, "a");
 if (fp1 == NULL || fp2 == NULL || fp3 == NULL)
 {
 puts("Could not open files");
 exit(0);
 }
 read(fp1,fp2,fp3);                      //user defined function to read and display
 printf("The %s and %s merged into %s file",a,b,c);
 fclose(fp1);
 fclose(fp2);                            //closing the files
 fclose(fp3);
 return 0;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9>notepad file1.txt

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9>notepad file2.txt

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9>notepad file3.txt

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9>gcc -c 1.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9>gcc 1.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9>a
Enter the 1st file name: file1.txt
Enter the 2nd file name: file2.txt
Enter the file name to merge contents of prev files: file3.txt
The file1.txt and file2.txt merged into file3.txt file
```



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

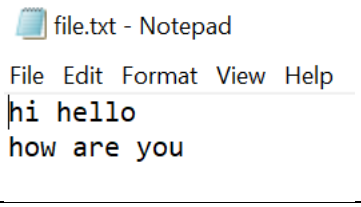
how are you

Program:

```
#include<stdio.h>
void read(char s[])
{ char str[100];
 int i,n;
 FILE *fp=fopen(s,"w+");
 printf(" Input the number of lines to be written : ");
 scanf("%d", &n);
 printf("\n :: The lines are ::\n");
 for(i=0;i<n+1;i++)
 {
 fgets(str,sizeof(str),stdin);          //takes input from terminal
 fputs(str,fp);                         //prints on the file entered
 }
 fclose(fp);
}
void printer(char s[])                  //prints the contents of the file
{ char str;
 FILE *fp=fopen(s,"r");
 printf("\n The content of the file %s is :\n",s);
 str = fgetc(fp);
 while (str != EOF)
 {
 printf ("%c", str);
 str = fgetc(fp);
 }
}
int main()
{
 char s[10];
 printf("Enter the filename:\n");       //enter the file_name
 gets(s);
 read(s);
 printer(s);
 printf("\n");
 return 0;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\2>gcc -c 2.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\2>gcc 2.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\2>a
Enter the filename:
file.txt
 Input the number of lines to be written : 2

 :: The lines are ::
hi hello
how are you

 The content of the file file.txt is :

hi hello
how are you


C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\2>type file.txt

hi hello
how are you
```



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 sort(int arr[],int size)                    //sort function
{
 int i,j,tmp;
 for(i=0; i<size; i++)
        for(j=i-1; j>=0; j--)
                if(arr[j]>arr[j+1])
                {
                        tmp=arr[j];
                        arr[j]=arr[j+1];
                        arr[j+1]=tmp;
```

```
                }
                else
                        break;
}
void printer(int a[100],int n)
{
 for(int i=0;i<n;++i)
        printf("%d ",a[i]);       //display the elements
}
int main()
{
 int n,a[100];
 printf("Enter the total no. of elements for sorting :\n");       //inputs the number of elements
 scanf("%d",&n);
 printf("Enter the elements of array:\n");
 for(int i=0;i<n;++i)                                             //enter the elements
        scanf("%d",&(a[i]));
 printf("Before sorting:\n");
 for(int i=0;i<n;++i)
         printf("%d ",a[i]);
 sort(a,n);                                                       //user-defined sort function
 printf("\nAfter sorting:\n");
 printer(a,n);
 return 0;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\3>gcc -c 3.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\3>gcc 3.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\3>a
Enter the total no. of elements for sorting :
5
Enter the elements of array:
13
6
23
1
81
Before sorting:
13 6 23 1 81
After sorting:
1 6 13 23 81
```

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> struct student //struct to store roll_no and name { int roll_no; char name[100];

```
};
typedef struct student STUDENT_T;
void display(STUDENT_T* p[],int n);
void swap(STUDENT_T **p,STUDENT_T **q);
void init_ptr(STUDENT_T s[],STUDENT_T* p[],int n);
void sort_data(STUDENT_T* p[],int n,int (*)(const STUDENT_T*,const STUDENT_T*));
int cmp_roll_no(const STUDENT_T*,const STUDENT_T*);
int cmp_name(const STUDENT_T*,const STUDENT_T*);

int main()
{
        FILE *f1 = fopen("student.csv","r");
        if(f1 == NULL)
        {
                perror("cannot open the file");
        }
        else
        {
                STUDENT_T st[1000];
                char line[100];
                fgets(line,100,f1);
                int i = 0;
                while(fgets(line,100,f1) != NULL)
                {
                        fputs(line,stdout);                     //displays the contents of the file
on terminal
                        char *r = strtok(line,",");             //store teh roll_no
                        char *name = strtok(NULL,",");                  //store the name
                        st[i].roll_no = atoi(r);          //converting string to int
                        strcpy(st[i].name,name);
                        i++;
                }
                int n = i;
                fclose(f1);
                STUDENT_T *p[1000];
                init_ptr(st,p,n);
                int ch;
                printf("enter your choice 1: sort on roll_no 2. sort on name\n");
                scanf("%d",&ch);
                switch(ch)
                {
                        case 1:sort_data(p,n,cmp_roll_no);display(p,n);break;  //sort based on
roll_no
```

```
                        case 2:sort_data(p,n,cmp_name);display(p,n);break; //sort based on
name
                        default: exit(0);
                }
        }
        return 0;
}
int cmp_roll_no(const STUDENT_T* s1,const STUDENT_T* s2)  //compare roll_no
{
                return s1->roll_no >s2->roll_no;
}
int cmp_name(const STUDENT_T* s1,const STUDENT_T* s2)  //compare name
{
        return strcmp(s1->name,s2->name)>0;
}
void init_ptr(STUDENT_T s[],STUDENT_T* p[],int n)
{
        int i;
        for(i = 0;i<n;i++)
                p[i] = &s[i];
}
void display(STUDENT_T* p[],int n)  //display
{
        int i;
        for(i = 0;i<n;i++)
                printf("%d %s\n",p[i]->roll_no,p[i]->name);
}
void sort_data(STUDENT_T* p[],int n,int (*op)(const STUDENT_T*, const
STUDENT_T*) )  //bubble sort
{
        int i,pos,j;
        for(i = 0;i<n-1;i++)
                for(j = 0;j<n-1;j++)
                        if(op(p[j],p[j+1]))
                                swap(&p[j],&p[j+1]);

}
void swap(STUDENT_T **p,STUDENT_T **q)   //swap function
{
        STUDENT_T *temp = *p;
        *p = *q;
        *q = temp;
}
```

Output Screenshot:

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\4>gcc -c 4.c

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\4>gcc 4.o

C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\4>a
99,xx
7,bb
22,cc
45,zz
8,aa
12,ff
4,gg
3,dd
27,jj
1,kk
32,ee
enter your choice 1: sort on roll_no 2. sort on name
1
1 kk

3 dd

4 gg

7 bb

8 aa

12 ff

22 cc

27 jj

32 ee

45 zz

99 xx
```

```
C:\Users\HP\Desktop\PESU\SEM-2\c programming lab\week 9\4>a
99,xx
7,bb
22,cc
45,zz
8,aa
12,ff
4,gg
3,dd
27,jj
1,kk
32,ee
enter your choice 1: sort on roll_no 2. sort on name
2
8 aa

7 bb

22 cc

3 dd

32 ee

12 ff

4 gg

27 jj

1 kk

99 xx

45 zz
```