Lab Assignment 8

Akshat Mittal - 20107

July 2021

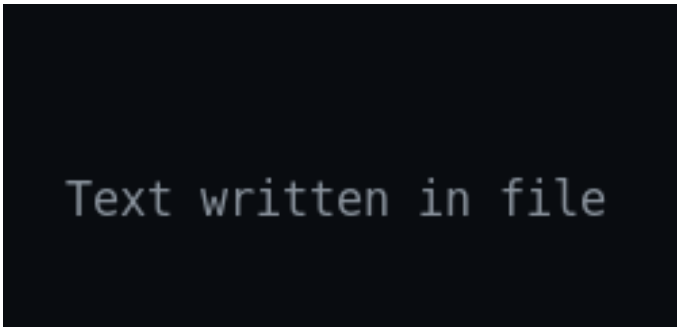
Contents

1

1.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    FILE *fio;
    fio = fopen("q1.txt", "w"); //q1.txt opened in write mode
    fprintf(fio, "%s\n%d", "Akshat Mittal", 7);
    fclose(fio); //data written and file closed
    printf(" Text written in file");
    printf("\n\n\n");
    return 0;
}
```

1.2 Output



1.3 Text File

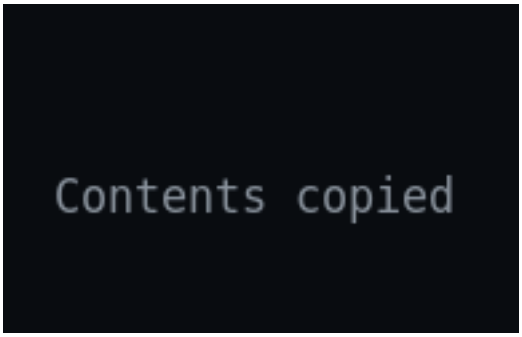
```
Akshat Mittal
7
```

2

2.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    FILE *fi, *fo;
    char ch;
    fi = fopen("q1.txt", "r"); //q1.txt opened in read mode
    fo = fopen("q2.txt", "w"); //q2.txt opened in write mode
    if (fi == NULL)
    {
        printf("Unable to locate file");
        return 0;
    }
    else
    {
        do
        {
            ch = getc(fi); //reading every character from q1.txt
            if (ch != EOF)
                fprintf(fo, "%c", ch); //writing character in q2.txt
        } while (ch != EOF);
        fclose(fi);
        fclose(fo);
        printf(" Contents copied");
    }
    printf("\n\n\n");
    return 0;
}
```

2.2 Output



2.3 Text File

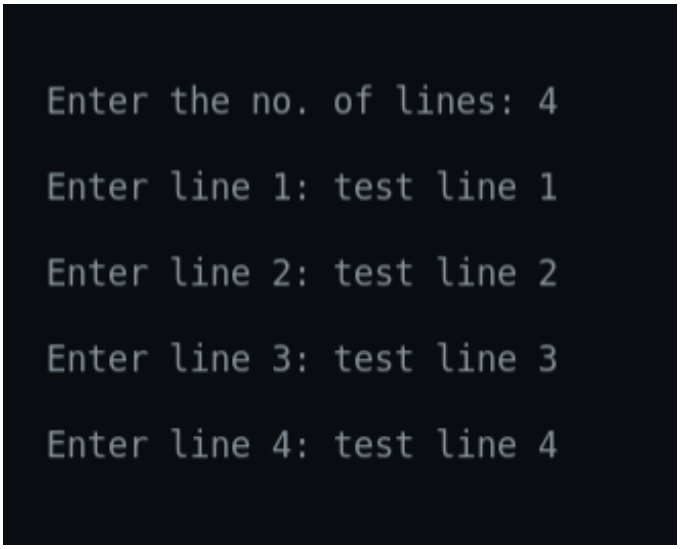
```
Akshat Mittal
7
```

3

3.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    int n, i;
    char lines[100][100], temp;
    FILE *fio;
    fio = fopen("q3.txt", "w"); //opening q3.txt in write mode
    printf("\n Enter the no. of lines: ");
    scanf("%d", &n);
    scanf("%c", &temp); //temporary variable to flush buffer
    for (i = 0; i < n; ++i)
    {
        printf("\n Enter line %d: ", i + 1);
        gets(lines[i]);
        fprintf(fio, "%s\n", lines[i]); //writing lines to q3.txt
    }
    printf("\n\n\n");
    return 0;
}
```

3.2 Output



3.3 Text File

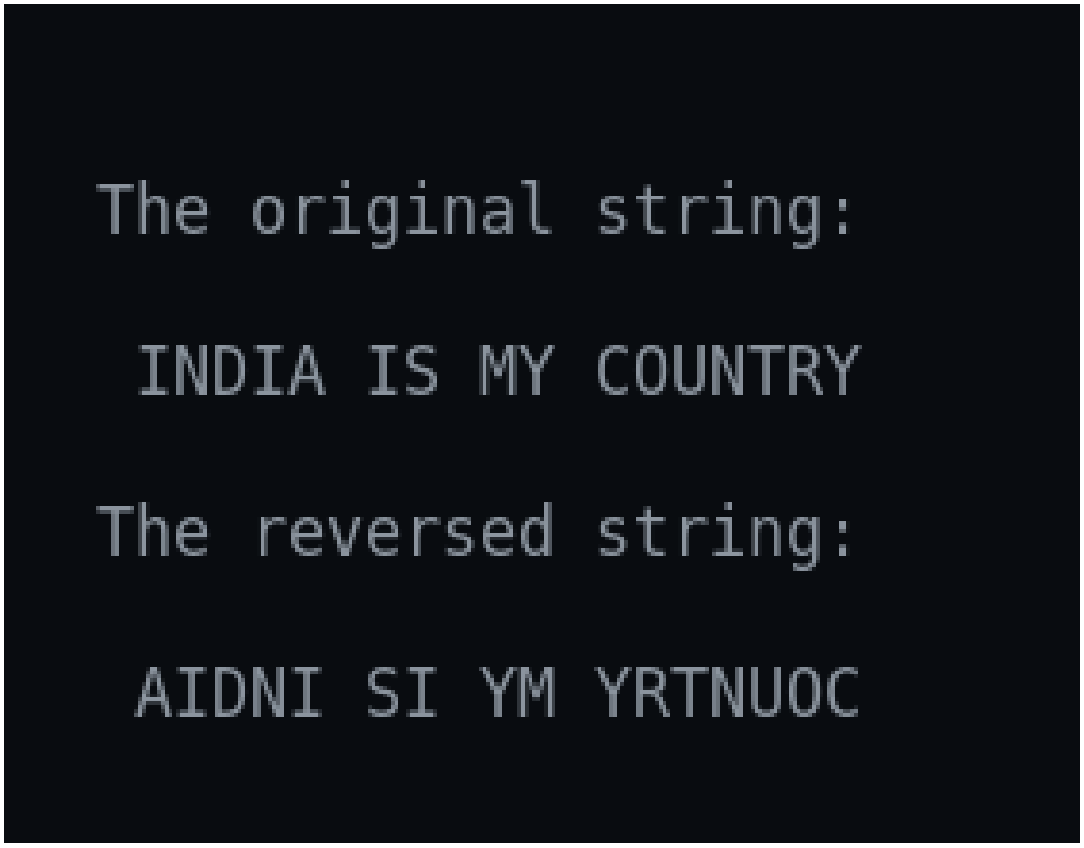
```
test line 1
test line 2
test line 3
test line 4
```

4

4.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    FILE *fio;
    int i = 0;
    char ch, str[20];
    fio = fopen("INPUT.TXT", "r"); //opening input.txt in read mode
    if (fio == NULL)
    {
        printf("Unable to locate the file");
        return 0;
    }
    else
    {
        do
        {
            ch = getc(fio); //getting file in string by character
            str[i++] = ch;
        } while (ch != EOF);
        str[i - 1] = '\0'; //to terminate string without EOF character
        fclose(fio);
    }
    int length = 0, j = 0, k;
    printf("\n\n The original string: \n\n %s", str);
    printf("\n\n The reversed string: \n\n ");
    for (i = 0; str[i] != '\0';)
    {
        for (length = j; str[length] != ' ' && str[length] != '\0'; ++length);
        if (i == 0)
            i = -1;
        ++i;
        for (int k = length - 1; k >= j && str[i] != '\0'; --k, ++i)
            printf("%c", str[k]); //printing the string from backwards
        j = length + 1;
        printf(" ");
    }
    printf("\n\n\n");
    return 0;
}
```

4.2 Output



4.3 Text File

```
INDIA IS MY COUNTRY
```

5

5.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    FILE *fio;
    int i = 0;
    char ch, str[1000];
    fio = fopen("TRIAL.TXT", "r"); //opening trial.txt in read mode
    if (fio == NULL)
    {
        printf("Unable to locate the file");
        return 0;
    }
    else
    {
        do
        {
            ch = getc(fio); //getting file in string by character
            if (ch == '\n')
                ch = ' '; //replacing new line feed by ' '
            str[i++] = ch;
        } while (ch != EOF);
        str[i - 1] = '\0'; //to terminate string without file terminator character
        fclose(fio);
    }
    int j = 0, k = 0, l = 0;
    printf("\n");
    printf("\n The string: \n%s", str);
    for (i = 0; str[i] != '\0'; ++i)
    {
        ++j;
        if (str[i] == ' ' || str[i + 1] == '\0')
        {
            ++k; //counting total words
            if ((j == 5) || (j == 4 && str[i + 1] == '\0'))
                ++l; //counting 4 letter words
            j = 0;
        }
    }
    printf("\n\n No. of words: %d", k);
    printf("\n\n 4 letter words: %d", l);
    printf("\n\n\n");
    return 0;
}
```

5.2 Output

```
 The string:
We keep this love in a photograph We made these memories for ourselves Where our eyes are never closing
 Hearts are never broken And time forever frozen still So you can keep me Inside the pocket of your rip
ped jeans Holding me closer until our eyes meet You wont ever be alone

 No. of words: 52

 4 letter words: 13
```

5.3 Text File

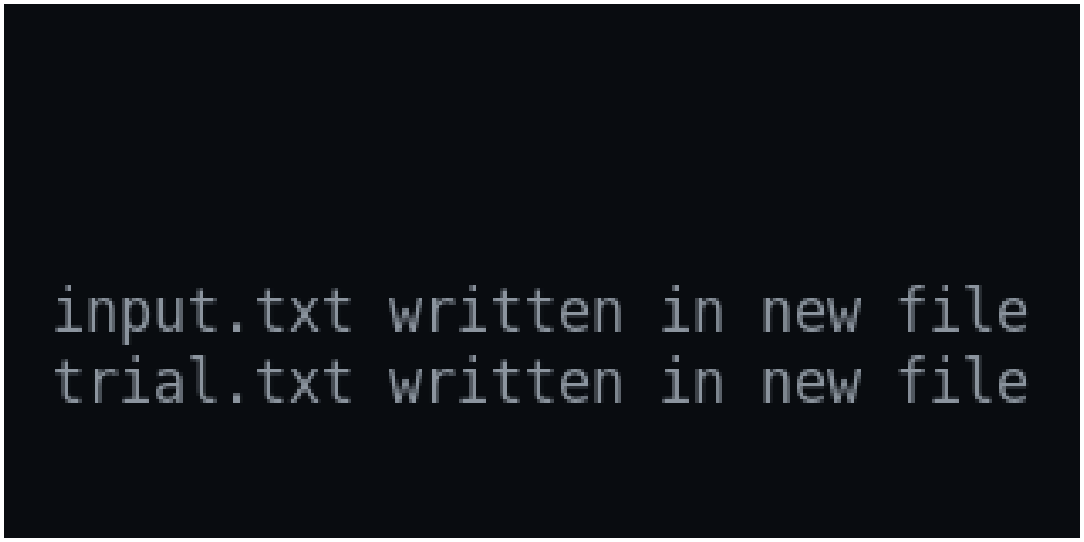
```
We keep this love in a photograph
We made these memories for ourselves
Where our eyes are never closing
Hearts are never broken
And time forever frozen still
So you can keep me
Inside the pocket of your ripped jeans
Holding me closer until our eyes meet
You wont ever be alone
```

6

6.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    FILE *fi1,*fi2, *fo;
    char ch;
    fi1 = fopen("TRIAL.TXT", "r");  //opening input.txt in read mode
    fi2 = fopen("INPUT.TXT", "r");  //opening trial.txt in read mode
    fo = fopen("q6.txt", "a");  //opening q6.txt in write mode
    if ((fi1 == NULL)||(fi2 == NULL))
    {
        printf("Unable to locate file");
        return 0;
    }
    else
    {
        do
        {
            ch = getc(fi1); //getting character print in fo
            fprintf(fo, "%c", ch);
        } while (ch != EOF);
        printf("\n input.txt written in new file");
        do
        {
            ch = getc(fi2); //getting character print in fo
            fprintf(fo, "%c", ch);
        } while (ch != EOF);
        printf("\n trial.txt written in new file");
        fclose(fi1);
        fclose(fi2);
        fclose(fo);
    }
    printf("\n\n\n");
    return 0;
}
```

6.2 Output



6.3 Text File

6.3.1 INPUT.TXT

```
INDIA IS MY COUNTRY
```

6.3.2 TRIAL.TXT

```
We keep this love in a photograph
We made these memories for ourselves
Where our eyes are never closing
Hearts are never broken
And time forever frozen still
So you can keep me
Inside the pocket of your ripped jeans
Holding me closer until our eyes meet
You wont ever be alone
```

7

7.1 Code

```
#include <stdio.h>
#include <string.h>
struct employee
{
    int number;
    char name[20];
    char sex;
    float gross_salary;
} Employee;
void print_employee(struct employee *employes)
{
    printf("\n Employee Code: %d", employes->number);
    printf("\n Employee Name: %s (%c)", employes->name, employes->sex);
    printf("\n Employee gross salary: %f", employes->gross_salary);
}
//declaring values for 5 employes
struct employee employes[5] = {{1, "Jonas", 'M', 7500},
                               {2, "Martha", 'F', 6000},
                               {3, "Klaus", 'M', 50000},
                               {4, "Stefen", 'M', 10000},
                               {5, "Caroline", 'F', 15000}};
int main()
{
    printf("\n\n\n");
    int i = 0, n = 0, choice, number;
    char ch = 'y', found = 'n', c, temp;
    float salary;
    struct employee employe[100];
    FILE *fio;
    /* fio = fopen("q7.dat", "w+b");
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (i < 5)
        {
            fwrite(&employes[i], sizeof(employes[i]), 1, fio);
            fflush(stdin);
            ++i;
        }
    }
    fclose(fio); */
    //code for writing default entries to q7.dat
```

```
print_all();
printf("\n 1. Enter details for new employee");
printf("\n 2. An employee is leaving");
printf("\n 3. An update in an employee's salary");
printf("\n 4. Print details of all employees");
do
{
    printf("\n\n Enter your choice: ");
    scanf("%d", &choice);
    switch (choice)
    {
    case 1:
        fio = fopen("q7.dat", "a+b"); //open q7.dat in append mode
        printf("\n");
        printf(" Employee Code: ");
        scanf("%d", &Employee.number);
        printf(" Employee Name: ");
        scanf("%s", Employee.name);
        scanf("%c", &temp);
        printf(" Employee sex: ");
        scanf("%c", &Employee.sex);
        printf(" Employee gross salary: ");
        scanf("%f", &Employee.gross_salary);
        //writing new entry in q7.dat
        fwrite(&Employee, sizeof(Employee), 1, fio);
        fflush(stdin);
        fclose(fio);
        printf("\n Record Added");
        break;
    case 2:
        n = 0;
        fio = fopen("q7.dat", "rb"); //open q7.dat in read mode
        if (fio == NULL)
        {
            printf("Cannot open file");
            return 0;
        }
        else
        {
            while (fread(&Employee, sizeof(Employee), 1, fio) == 1)
            {
                employe[n].number = Employee.number;
                strcpy(employe[n].name, Employee.name);
                employe[n].sex = Employee.sex;
                employe[n].gross_salary = Employee.gross_salary;
                ++n;
            }
        }
        fclose(fio);
        printf("\n Enter the serial number of employee: ");
```

```
    scanf("%d", &number);
    fio = fopen("q7.dat", "w+b"); //open q7.dat in write mode
    i = 0;
    while (i < n)
    {
        if (employe[i].number == number)
            employe[i].gross_salary = 0;
        //rewrite every entry except for removed employee salary=0
        fwrite(&employe[i], sizeof(employe[i]), 1, fio);
        fflush(stdin);
        ++i;
    }
    fclose(fio);
    printf("\n Record Deleted");
    break;
case 3:
    n = 0;
    fio = fopen("q7.dat", "rb"); //open q7.dat in read mode
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (fread(&Employee, sizeof(Employee), 1, fio) == 1)
        {
            employe[n].number = Employee.number;
            strcpy(employe[n].name, Employee.name);
            employe[n].sex = Employee.sex;
            employe[n].gross_salary = Employee.gross_salary;
            ++n;
        }
    }
    fclose(fio);
    printf(" Enter the serial number of employee: ");
    scanf("%d", &number);
    printf(" Enter new salary: ");
    scanf("%f", &salary);
    fio = fopen("q7.dat", "w+b"); //open q7.dat in write mode
    i = 0;
    while (i < n)
    {
        if (employe[i].number == number)
            employe[i].gross_salary = salary;
        //rewrite every entry except for modified employee salary
        fwrite(&employe[i], sizeof(employe[i]), 1, fio);
        fflush(stdin);
        ++i;
    }
```

```
            fclose(fio);
            printf("\n Record Modified");
            break;
        case 4:
            print_all();
            break;
        default:
            printf("\n Wrong choice");
            break;
        }
        scanf("%c", &temp);
        printf("\n Again? ");
        scanf("%c", &ch);
    } while (ch == 'y' || ch == 'Y');
    printf("\n\n\n");
    return 0;
}
int print_all()
{
    FILE *fio;
    fio = fopen("q7.dat", "rb"); //open q7.dat in read mode
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (fread(&Employee, sizeof(Employee), 1, fio) == 1)
        {
            if (Employee.gross_salary != 0)
            {
                print_employee(&Employee);
                printf("\n");
            }
        }
    }
    fclose(fio);
    return 0;
}
```

7.2 Output

```
Employee Code: 1
Employee Name: Jonas (M)
Employee gross salary: 7500.000000

Employee Code: 2
Employee Name: Martha (F)
Employee gross salary: 6000.000000

Employee Code: 3
Employee Name: Klaus (M)
Employee gross salary: 50000.000000

Employee Code: 4
Employee Name: Stefen (M)
Employee gross salary: 10000.000000

Employee Code: 5
Employee Name: Caroline (F)
Employee gross salary: 15000.000000

1. Enter details for new employee
2. An employee is leaving
3. An update in an employee's salary
4. Print details of all employees
```

```
1. Enter details for new employee
2. An employee is leaving
3. An update in an employee's salary
4. Print details of all employees

Enter your choice: 1

Employee Code: 6
Employee Name: Elijah
Employee sex: M
Employee gross salary: 40000

Record Added
Again? y


Enter your choice: 2

Enter the serial number of employee: 4

Record Deleted
Again? y


Enter your choice: 3
Enter the serial number of employee: 2
Enter new salary: 5000

Record Modified
Again? y


Enter your choice: 4
```

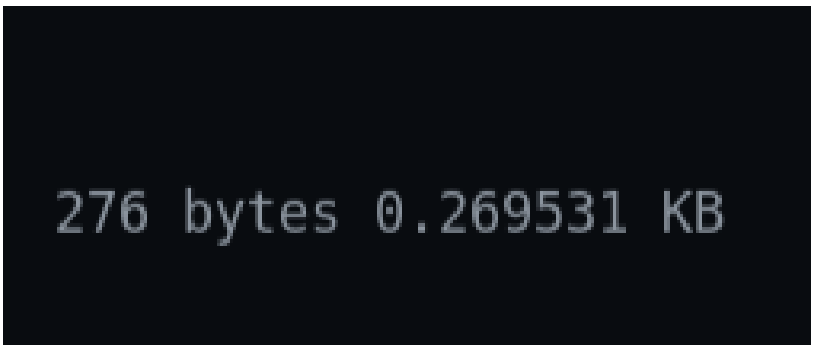
```
Enter your choice: 4

Employee Code: 1
Employee Name: Jonas (M)
Employee gross salary: 7500.000000

Employee Code: 2
Employee Name: Martha (F)
Employee gross salary: 5000.000000

Employee Code: 3
Employee Name: Klaus (M)
Employee gross salary: 50000.000000

Employee Code: 5
Employee Name: Caroline (F)
Employee gross salary: 15000.000000

Employee Code: 6
Employee Name: Elijah (M)
Employee gross salary: 40000.000000

Again? n
```

8

8.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    FILE *fio;
    fio = fopen("TRIAL.TXT", "r"); //open trial.txt in read mode
    if (!fio)
        printf("Cannot open");
    else
    {
        fseek(fio, 0, SEEK_END); //to put file pointer at the end
        printf("%ld bytes %f KB", ftello(fio), (float)ftello(fio) / 1024);
    }
    fclose(fio);
    printf("\n\n\n");
    return 0;
}
```

8.2 Output



8.3 Text File

```
We keep this love in a photograph
We made these memories for ourselves
Where our eyes are never closing
Hearts are never broken
And time forever frozen still
So you can keep me
Inside the pocket of your ripped jeans
Holding me closer until our eyes meet
You wont ever be alone
```

9

9.1 Code

```
#include <stdio.h>
#include <string.h>
struct student
{
    char name[20];
    int age;
} Student;
struct student stud[5] = {{"Jonas", 20},
                          {"Martha", 20},
                          {"Klaus", 19},
                          {"Stefen", 17},
                          {"Caroline", 17}};
int main()
{
    printf("\n\n\n");
    /* int i = 0;
    FILE *fio;
    fio = fopen("q9.dat", "w+b");
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (i<5)
        {
            fwrite(&stud[i], sizeof(stud[i]), 1, fio);
            fflush(stdin);
            ++i;
        }
    }
    fclose(fio); */
    //code for writing default entries to q7.dat
    print_sort();
    printf("\n\n\n");
    return 0;
}
int print_sort()
{
    struct student Stud[100], temp;
    int n = 0;
    FILE *fio;
    fio = fopen("q9.dat", "rb"); //open q9.dat in read mode
    if (fio == NULL)
    {
```

```
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (fread(&Student, sizeof(Student), 1, fio) == 1)
        {
            strcpy(Stud[n].name, Student.name);
            Stud[n].age = Student.age;
            ++n; //counting number of students
        }
        for (int i = 0; i < n; ++i)
        {
            for (int j = i + 1; j < n; ++j)
            {
                if (Stud[i].name[0] > Stud[j].name[0]) //arranging by bubble sort
                {
                    temp = Stud[i];
                    Stud[i] = Stud[j];
                    Stud[j] = temp;
                }
            }
        }
        for (int i = 0; i < n; ++i)
        {
            printf("\n Name: %s", Stud[i].name);
            printf("\n Age: %d", Stud[i].age);
            printf("\n");
        }
    }
    fclose(fio);
    return 0;
}
```

9.2 Output

```
Name: Caroline
Age: 17

Name: Jonas
Age: 20

Name: Klaus
Age: 19

Name: Martha
Age: 20

Name: Stefen
Age: 17
```

10

10.1 Code

```c
#include <stdio.h>
int main()
{
    FILE *fi1, *fi2, *fo;
    char ch, ch1 = ' ', ch2 = '\n';
    fi1 = fopen("q10a.txt", "r"); //opening q10a.txt in read mode
    fi2 = fopen("q10b.txt", "r"); //opening q10b.txt in read mode
    fo = fopen("q10.txt", "w");   //opening q10.txt in write mode
    if ((fi1 == NULL) || (fi2 == NULL))
    {
        printf("Unable to locate file");
        return 0;
    }
    else
    {
        do
        {
            if (ch1 != EOF)
            {
                do
                {
                    ch1 = getc(fi1); //write characters until \n
                    if (ch1 == EOF)
                        break;
                    fprintf(fo, "%c", ch1);
                } while (ch1 != '\n');
            }
            if (ch2 != EOF)
            {
                do
                {
                    ch2 = getc(fi2); //write characters until \n
                    if (ch2 == EOF)
                        break;
                    fprintf(fo, "%c", ch2);
                } while (ch2 != '\n');
            }
        } while (ch2 != EOF || ch1 != EOF);
        printf("\n Files copied alternatively");
        fclose(fi1);
        fclose(fi2);
    }
    fclose(fo);
    printf("\n\n\n");
    return 0;
}
```

10.2 Output

10.3 Text File

10.3.1 q10a.txt

```
Who says, who says youre not perfect?
Who says youre not worth it?
Who says youre the only one thats hurting?
Trust me, thats the price of beauty
Who says youre not pretty?
Who says youre not beautiful?
```

10.3.2 q10b.txt

```
We keep this love in a photograph
We made these memories For ourselves
Where our eyes are never closing
Hearts are never broken, and time forever frozen still
So you can keep me
Inside the pocket of your ripped jeans
Holding me closer until our eyes meet
You wont ever be alone
```

10.3.3 q10.txt

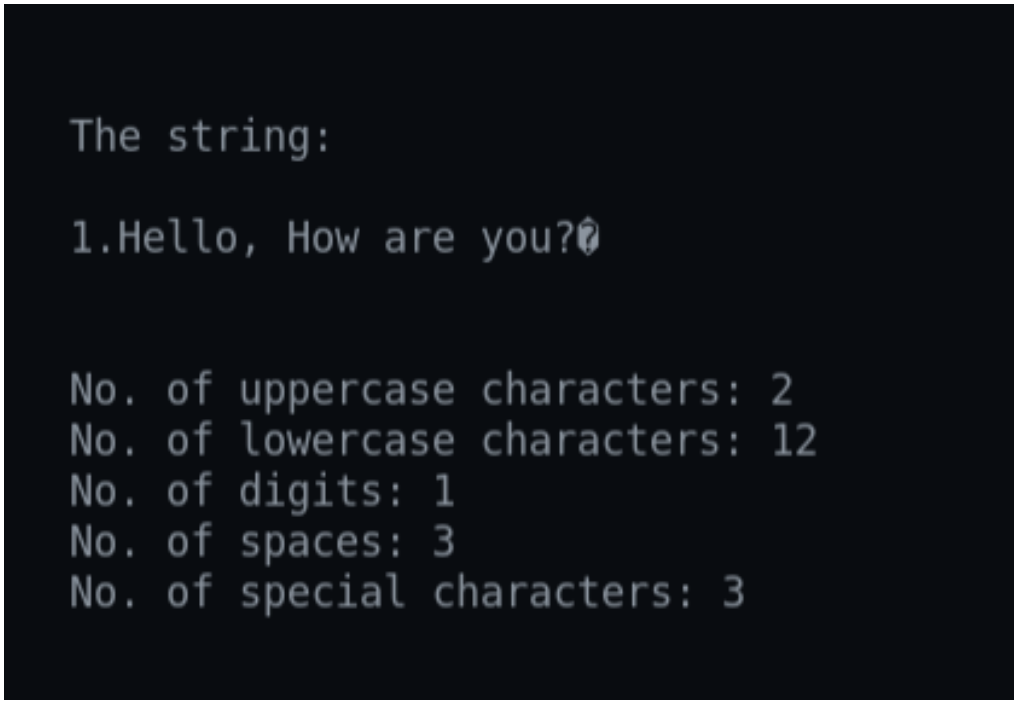
```
Who says, who says youre not perfect?
We keep this love in a photograph
Who says youre not worth it?
We made these memories For ourselves
Who says youre the only one thats hurting?
Where our eyes are never closing
Trust me, thats the price of beauty
Hearts are never broken, and time forever frozen still
Who says youre not pretty?
So you can keep me
Who says youre not beautiful?
Inside the pocket of your ripped jeans
Holding me closer until our eyes meet
You wont ever be alone
```

11

11.1 Code

```c
#include <stdio.h>
int main()
{
    printf("\n\n\n");
    int lower = 0, upper = 0, number = 0, spaces = 0, special = -1;
    //special=-1 to neglect last terminating character provided by file pointer
    char ch;
    FILE *fio;
    fio = fopen("q11.txt", "r"); //open q11.txt in read mode
    printf("\n The string: \n\n ");
    do
    {
        ch = fgetc(fio);
        printf("%c", ch);
        if (ch >= '0' && ch <= '9')
            ++number; //for digits
        else if (ch >= 'A' && ch <= 'Z')
            ++upper; //for upppercase characters
        else if (ch >= 'a' && ch <= 'z')
            ++lower; //for upppercase characters
        else if (ch == ' ')
            ++spaces; //for spaces
        else
            ++special; //for special characters
    } while (ch != EOF);
    fclose(fio);
    printf("\n\n");
    printf("\n No. of uppercase characters: %d", upper);
    printf("\n No. of lowercase characters: %d", lower);
    printf("\n No. of digits: %d", number);
    printf("\n No. of spaces: %d", spaces);
    printf("\n No. of special characters: %d", special);
    printf("\n\n\n");
    return 0;
}
```

11.2 Output



11.3 Text File

```
1.Hello, How are you?
```

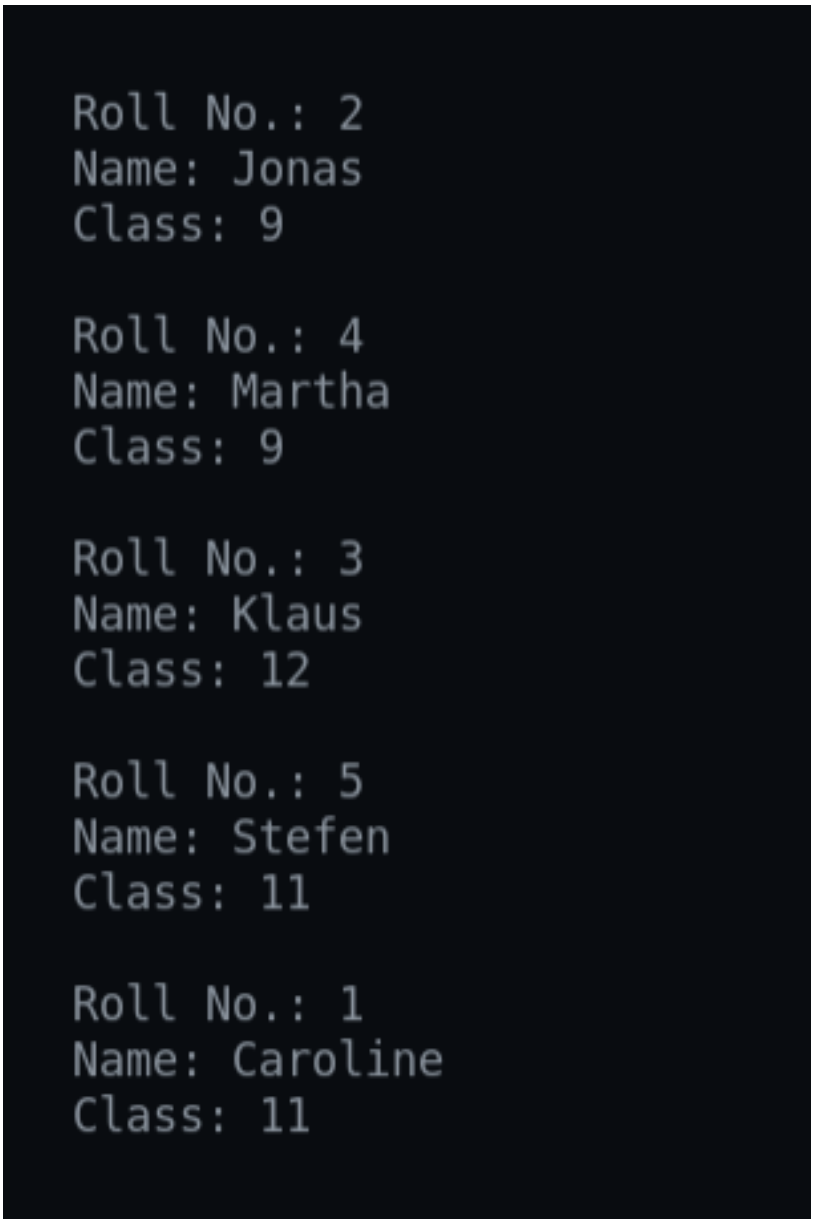
12

12.1 Code

```
#include <stdio.h>
#include <string.h>
struct student
{
    int roll;
    char name[20];
    int Class;
} Student;
struct student stud[5] = {{2, "Jonas", 9},
                          {4, "Martha", 9},
                          {3, "Klaus", 12},
                          {5, "Stefen", 11},
                          {1, "Caroline", 11}};
//declaring values for 5 students
int main()
{
    printf("\n\n\n");
    int i = 0;
    FILE *fio;
    fio = fopen("q12.dat", "w+b"); //open q12.dat in write mode
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (i < 5)
        {
            fwrite(&stud[i], sizeof(stud[i]), 1, fio); //writing records to file
            fflush(stdin);
            ++i;
        }
    }
    fclose(fio);
    fio = fopen("q12.dat", "rb"); //open q12.dat in read mode
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (fread(&Student, sizeof(Student), 1, fio) == 1)
        {
            printf("\n Roll No.: %d", Student.roll);
```

```
            printf("\n Name: %s", Student.name);
            printf("\n Class: %d", Student.Class);
            printf("\n");
        }
    }
    fclose(fio);
    printf("\n\n\n");
    return 0;
}
```

12.2 Output



13

13.1 Code

```
#include <stdio.h>
#include <string.h>
struct student
{
    int roll;
    char name[20];
    int Class;
} Student;
struct student stud[5] = {{2, "Jonas", 9},
                          {4, "Martha", 9},
                          {3, "Klaus", 12},
                          {5, "Stefen", 11},
                          {1, "Caroline", 11}};
//declaring values for 5 students
void print_student(struct student *stud)
{
    printf("\n Roll No.: %d", stud->roll);
    printf("\n Name: %s", stud->name);
    printf("\n Class: %d", stud->Class);
}
int main()
{
    printf("\n\n\n");
    int i = 0, n = 0, choice, number;
    char ch = 'y', found = 'n', c, temp;
    float salary;
    struct student Stud[100];
    FILE *fio;
    /* fio = fopen("q13.dat", "w+b");
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (i < 5)
        {
            fwrite(&stud[i], sizeof(stud[i]), 1, fio);
            fflush(stdin);
            ++i;
        }
    }
    fclose(fio); */
    //code for writing default entries to q13.dat
    print_all();
```

```
printf("\n 1. Enter details for new student");
printf("\n 2. Details of a particular student");
printf("\n 3. Edit a student's details");
printf("\n 4. Remove a student");
printf("\n 5. Print details of all students");
do
{
    printf("\n\n Enter your choice: ");
    scanf("%d", &choice);
    switch (choice)
    {
    case 1:
        fio = fopen("q13.dat", "a+b"); //open q13.dat in append mode
        printf("\n");
        printf(" Student Roll No.: ");
        scanf("%d", &Student.roll);
        printf(" Student Name: ");
        scanf("%s", Student.name);
        scanf("%c", &temp);
        printf(" Student Class: ");
        scanf("%d", &Student.Class);
        //writing new entry in q13.dat
        fwrite(&Student, sizeof(Student), 1, fio);
        fflush(stdin);
        fclose(fio);
        printf("\n Record Added");
        break;
    case 2:
        n = 0, i = 0;
        fio = fopen("q13.dat", "rb"); //open q13.dat in read mode
        if (fio == NULL)
        {
            printf("Cannot open file");
            return 0;
        }
        else
        {
            printf("\n Enter the roll number of student: ");
            scanf("%d", &number);
            while (fread(&Student, sizeof(Student), 1, fio) == 1)
            {
                if (Student.roll == number)
                {
                    print_student(&Student);
                    i = 1;
                }
            }
            if (!i)
                printf("\n No Student found");
        }
```

```
    fclose(fio);
    break;
case 3:
    n = 0;
    fio = fopen("q13.dat", "rb"); //open q13.dat in read mode
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (fread(&Student, sizeof(Student), 1, fio) == 1)
        {
            Stud[n].roll = Student.roll;
            strcpy(Stud[n].name, Student.name);
            Stud[n].Class = Student.Class;
            ++n;
        }
    }
    fclose(fio);
    printf(" Enter the roll number of student: ");
    scanf("%d", &number);
    fio = fopen("q13.dat", "w+b"); //open q13.dat in write mode
    i = 0;
    while (i < n)
    {
        if (Stud[i].roll == number)
        {
            printf("\n");
            printf(" Student Roll No.: ");
            scanf("%d", &Stud[i].roll);
            printf(" Student Name: ");
            scanf("%s", Stud[i].name);
            scanf("%c", &temp);
            printf(" Student Class: ");
            scanf("%d", &Stud[i].Class);
        }
        //rewrite every entry except for modified student
        fwrite(&Stud[i], sizeof(Stud[i]), 1, fio);
        fflush(stdin);
        ++i;
    }
    fclose(fio);
    printf("\n Record Modified");
    break;
case 4:
    n = 0;
    fio = fopen("q13.dat", "rb"); //open q13.dat in read mode
    if (fio == NULL)
```

```
            {
                printf("Cannot open file");
                return 0;
            }
            else
            {
                while (fread(&Student, sizeof(Student), 1, fio) == 1)
                {
                    Stud[n].roll = Student.roll;
                    strcpy(Stud[n].name, Student.name);
                    Stud[n].Class = Student.Class;
                    ++n;
                }
            }
            fclose(fio);
            printf(" Enter the roll number of student: ");
            scanf("%d", &number);
            fio = fopen("q13.dat", "w+b"); //open q13.dat in write mode
            i = 0;
            while (i < n)
            {
                if (Stud[i].roll != number)
                {
                    fwrite(&Stud[i], sizeof(Stud[i]), 1, fio);
                    fflush(stdin);
                }
                //rewrite every entry except for removed student
                ++i;
            }
            fclose(fio);
            printf("\n Record Deleted");
            break;
        case 5:
            print_all();
            break;
        default:
            printf("\n Wrong choice");
            break;
        }
        scanf("%c", &temp);
        printf("\n Again? ");
        scanf("%c", &ch);
    } while (ch == 'y' || ch == 'Y');
    printf("\n\n\n");
    return 0;
}
int print_all()
{
    FILE *fio;
    fio = fopen("q13.dat", "rb"); //open q13.dat in read mode
```

```
    if (fio == NULL)
    {
        printf("Cannot open file");
        return 0;
    }
    else
    {
        while (fread(&Student, sizeof(Student), 1, fio) == 1)
        {
            if (Student.roll != 0)
            {
                print_student(&Student);
                printf("\n");
            }
        }
    }
    fclose(fio);
    return 0;
}
```

13.2 Output

```
Roll No.: 2
Name: Jonas
Class: 9

Roll No.: 4
Name: Martha
Class: 9

Roll No.: 3
Name: Klaus
Class: 12

Roll No.: 5
Name: Stefen
Class: 11

Roll No.: 1
Name: Caroline
Class: 11

1. Enter details for new student
2. Details of a particular student
3. Edit a student's details
4. Remove a student
5. Print details of all students

Enter your choice: 1

Student Roll No.: 6
Student Name: Elijah
Student Class: 12
```

```
Enter your choice: 1

Student Roll No.: 6
Student Name: Elijah
Student Class: 12

Record Added
Again? y


Enter your choice: 2

Enter the roll number of student: 3

Roll No.: 3
Name: Klaus
Class: 12
Again? y


Enter your choice: 3
Enter the roll number of student: 4

Student Roll No.: 4
Student Name: Cami
Student Class: 10

Record Modified
Again? y
```

```
Enter your choice: 4
Enter the roll number of student: 2

Record Deleted
Again? y


Enter your choice: 5

Roll No.: 4
Name: Cami
Class: 10

Roll No.: 3
Name: Klaus
Class: 12

Roll No.: 5
Name: Stefen
Class: 11

Roll No.: 1
Name: Caroline
Class: 11

Roll No.: 6
Name: Elijah
Class: 12

Again? n
```