Green University of Bangladesh
Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering
Semester: (Spring, Year:2021), B.Sc. in CSE (Day/Eve)

Course Title:CSE Lab
Course Code: CSE 106 Section:DD

Lab Project Name: BANK MANAGEMENT SYSTEM

Student Details

Name	ID
Arifuzzaman Fahim	**213902039**

Submission Date: 9/10/2022
Course Teacher's Name: Md. Sultanul Islam Ovi

[For Teachers use only: Don't Write Anything inside this box]

Lab Project Status

Marks: ……	**Signature: .....**
Comments:	**Date: .....**

Table of Contents

Chapter 1

Introduction

1.1 Introduction

The Bank Management System (BMS) is **a web-based application used for paying financial institutions for the services they provide to the Bureau of the Fiscal Service**. BMS also provides analytical tools to review, and approve compensation, budgets, and outflows.

1.2 Design Goals/Objective

The main objective of bank management is to build organic and optimal system of interaction between the elements of banking mechanism with a view to profit. Successful optimization of the "profitability-risk" ratio in a bank lending operations is largely determined by the use of effective methods of bank management.

Chapter 2

Design/Development/Implementation of the Project

2.1 Front Screen Design

2.2 PSEUDO CODE

HEADER FILES & VARIABLES

```
//Arifuzzaman Fahim
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <stdbool.h>
#define TOTAL_user 100
typedef struct
{
   char name[50];
   int age;
   char address[20];
   double balance;
} User;
```

Adding User

```
void add_User(User user[])
{
   file1 = fopen("file.txt", "w+");
   char another = 'y';
   while (another == 'y')
   {
      printf("\nUser %d info:\n", record + 1);
      fflush(stdin);
      printf("Enter Name: ");
      scanf("%[^\n]%*c", user[record].name);
      fflush(stdin);
      printf("Enter age: ");
      scanf("%d", &user[record].age);
      fflush(stdin);
      printf("Enter Address: ");
```

```
        scanf("%[^\n]%*c", user[record].address);
        fflush(stdin);
        printf("Enter Blance: ");
        fflush(stdin);
        scanf("%lf", &user[record].balance);
        printf("\n");
        printf("Do you want to Add another user? (y/n): ");
        fflush(stdin);
        another = getchar();
        record++;
    }
    for (i = 0; i < record; i++)
    {
        fprintf(file1, "%s,%d,%s,%.2lf\n", user[i].name, user[i].age, user[i].address,
user[i].balance);
        if (ferror(file1))
        {
            printf("Error writing to file1\n");
            exit(1);
        }
    }
    fclose(file1);
    printf("\n%d User added to file1\n", record);
}
```

List function

```
void list_User(User Users[])
{
    printf("%-7s %-20s %-7s %-20s %-10s\n", "Serial", "Name", "age", "Address",
"Balance");
    for (i = 0; i < record; i++)
    {
        printf("%-7d %-20s %-7d %-20s %-10.2lf\n", i, user[i].name, user[i].age,
user[i].address, user[i].balance);
    }
    system("pause");
}
```

User Modify

```
void modify_User(User user[])
{
   printf("Enter the serial number of the User you want to modify: ");
   fflush(stdin);
   scanf("%d", &n);
   printf("\nUser %d info:\n", n);
   fflush(stdin);
   printf("Name: ");
   scanf("%[^\n]%*c", user[n].name);
   fflush(stdin);
   printf("age: ");
   scanf("%d", &user[n].age);
   fflush(stdin);
   printf("Address: ");
   scanf("%[^\n]%*c", user[n].address);
   fflush(stdin);
   printf("Balance: ");
   fflush(stdin);
   scanf("%lf", &user[n].balance);
   printf("\n");
   file1 = fopen("CCsvf.csv", "w+");
   for (i = 0; i < record; i++)
   {
      fprintf(file1, "%s,%d,%s,%lf\n", user[i].name, user[i].age, user[i].address,
user[i].balance);
      if (ferror(file1))
      {
         printf("Error writing to file1\n");
         exit(1);
      }
   }
   printf("User %d modified\n", n);
   fclose(file1);
   system("pause");
}
```

Delete

```
void delete_User(User user[])
{
   printf("Enter the User serial number : ");
   fflush(stdin);
   scanf("%d", &n);
   for (i = n; i < record; i++)
   {
      strcpy(user[i].name, user[i + 1].name);
      user[i].age = user[i + 1].age;
      strcpy(user[i].address, user[i + 1].address);
      user[i].balance = user[i + 1].balance;
   }
   record--;
   file1 = fopen("CCsvf.csv", "w+");
   for (i = 0; i < record; i++)
   {
      fprintf(file1, "%s,%d,%s,%.2lf\n", user[i].name, user[i].age, user[i].address,
user[i].balance);
      if (ferror(file1))
      {
         printf("Error writing to file1\n");
         exit(1);
      }
   }
   fclose(file1);
   printf("User %d is deleted\n", n);
   system("pause");
}
```

Search

```
void search_User(User user[])
{

   printf("Enter the name of the User you want to search: ");
   fflush(stdin);
   char name[40];
   scanf("%[^\n]%*c", name);
   for (i = 0; i < record; i++)
   {
```

```
      if (strcmp(user[i].name, name) == 0)
      {
         printf("%-7s %-20s %-7s %-20s %-10s\n", "Serial", "Name", "age",
"Address", "Balance");
         printf("%-7d %-20s %-7d %-20s %-10.2lf\n", i, user[i].name, user[i].age,
user[i].address, user[i].balance);
         system("pause");
      }
   }
}
```

Sorting

```
void sort_User_name(User user[])
{
   int i, j;
   User tuser;
   for (i = 0; i < record; i++)
   {
      for (j = 0; j < record - 1; j++)
      {
         if (strcmp(user[j].name, user[j + 1].name) > 0)
         {
            strcpy(tuser.name, user[j].name);
            tuser.age = user[j].age;
            strcpy(tuser.address, user[j].address);
            tuser.balance = user[j].balance;
            strcpy(user[j].name, user[j + 1].name);
            user[j].age = user[j + 1].age;
            strcpy(user[j].address, user[j + 1].address);
            user[j].balance = user[j + 1].balance;
            strcpy(user[j + 1].name, tuser.name);
            user[j + 1].age = tuser.age;
            strcpy(user[j + 1].address, tuser.address);
            user[j + 1].balance = tuser.balance;
         }
      }
   }
   file1 = fopen("CCsvf.csv", "w+");
   for (i = 0; i < record; i++)
   {
```

```
      fprintf(file1, "%s,%d,%s,%.2lf\n", user[i].name, user[i].age, user[i].address,
user[i].balance);
      if (ferror(file1))
      {
         printf("Error writing to file1\n");
         exit(1);
      }
   }
   fclose(file1);
   printf("User sorted by name done successfully\n");
   system("pause");
}
void sort_User_age(User user[])
{
   int i, j;
   User tuser;
   for (i = 0; i < record; i++)
   {
      for (j = 0; j < record - 1; j++)
      {
         if (user[j].age > user[j + 1].age)
         {
            strcpy(tuser.name, user[j].name);
            tuser.age = user[j].age;
            strcpy(tuser.address, user[j].address);
            tuser.balance = user[j].balance;
            strcpy(user[j].name, user[j + 1].name);
            user[j].age = user[j + 1].age;
            strcpy(user[j].address, user[j + 1].address);
            user[j].balance = user[j + 1].balance;
            strcpy(user[j + 1].name, tuser.name);
            user[j + 1].age = tuser.age;
            strcpy(user[j + 1].address, tuser.address);
            user[j + 1].balance = tuser.balance;
         }
      }
   }
   file1 = fopen("CCsvf.csv", "w+");
   for (i = 0; i < record; i++)
   {
      fprintf(file1, "%s,%d,%s,%.2lf\n", user[i].name, user[i].age, user[i].address,
user[i].balance);
```

```
        if (ferror(file1))
        {
            printf("Error writing to file1\n");
            exit(1);
        }
    }
    fclose(file1);
    printf("User sorted by age successfully\n");
    system("pause");
}

void sort_User_address(User user[])
{
    int i, j;
    User tuser;
    for (i = 0; i < record; i++)
    {
        for (j = 0; j < record - 1; j++)
        {
            if (strcmp(user[j].address, user[j + 1].address) > 0)
            {
                strcpy(tuser.name, user[j].name);
                tuser.age = user[j].age;
                strcpy(tuser.address, user[j].address);
                tuser.balance = user[j].balance;
                strcpy(user[j].name, user[j + 1].name);
                user[j].age = user[j + 1].age;
                strcpy(user[j].address, user[j + 1].address);
                user[j].balance = user[j + 1].balance;
                strcpy(user[j + 1].name, tuser.name);
                user[j + 1].age = tuser.age;
                strcpy(user[j + 1].address, tuser.address);
                user[j + 1].balance = tuser.balance;
            }
        }
    }
    file1 = fopen("CCsvf.csv", "w+");
    for (i = 0; i < record; i++)
    {
        fprintf(file1, "%s,%d,%s,%.2lf\n", user[i].name, user[i].age, user[i].address,
user[i].balance);
        if (ferror(file1))
```

```
      {
         printf("Error writing to file1\n");
         exit(1);
      }
   }
   fclose(file1);
   printf("User is sorted by address successfully\n");
   system("pause");
}
void sort_user_age(User user[])
{
   int i, j;
   User tuser;
   for (i = 0; i < record; i++)
   {
      for (j = 0; j < record - 1; j++)
      {
         if (user[j].balance > user[j + 1].balance)
         {
            strcpy(tuser.name, user[j].name);
            tuser.age = user[j].age;
            strcpy(tuser.address, user[j].address);
            tuser.balance = user[j].balance;
            strcpy(user[j].name, user[j + 1].name);
            user[j].age = user[j + 1].age;
            strcpy(user[j].address, user[j + 1].address);
            user[j].balance = user[j + 1].balance;
            strcpy(user[j + 1].name, tuser.name);
            user[j + 1].age = tuser.age;
            strcpy(user[j + 1].address, tuser.address);
            user[j + 1].balance = tuser.balance;
         }
      }
   }
   file1 = fopen("CCsvf.csv", "w+");
   for (i = 0; i < record; i++)
   {
      fprintf(file1, "%s,%d,%s,%lf\n", user[i].name, user[i].age, user[i].address,
user[i].balance);
      if (ferror(file1))
      {
         printf("Error writing to file1\n");
```

```
            exit(1);
        }
    }
    fclose(file1);
    printf("User sorted by balance successfully\n");
    system("pause");
}
```

MAIN FUNCTION

```
int main()
{

    fileload(user);

    while (1)
    {
        printf(" \n\t\t\tWELLCOME TO Fahim BANK MANAGEMENT SYSTEM \n");
        printf("  \t\t\t\t 1. Add User Records \n");
        printf("  \t\t\t\t 2. List User Records \n");
        printf("  \t\t\t\t 3. Modify User Records\n");
        printf("  \t\t\t\t 4. Delete User Records\n");
        printf("  \t\t\t\t 5. Search User Records\n");
        printf("  \t\t\t\t 6. Sort User Records\n");
        printf("  \t\t\t\t 7. Exit System\n");
        printf("
_____
_________\n");
        printf("
_____
_________");
        printf(" \n \n Your Choice: ");
        fflush(stdin);
        choice = getchar();
        switch (choice)
        {
        case '1':
            system("cls");
            add_User(user);
```

```
        break;
    case '2':
        system("cls");
        list_User(user);
        break;
    case '3':
        system("cls");
        modify_User(user);
        break;
    case '4':
        system("cls");
        delete_User(user);
        break;
    case '5':
        system("cls");
        search_User(user);

        break;
    case '6':
        system("cls");

        printf(" \n\t\t\tBANK MANageMENT SYSTEM\n");
        printf("1. Sort by Name \n");
        printf("2. Sort by age \n");
        printf("3. Sort by Address \n");
        printf("4. Sort by Basic age \n");
        printf("5. Return Back \n");
        printf(" \n \n Your Choice: ");
        fflush(stdin);
        choice = getchar();
        switch (choice)
        {
        case '1':
            system("cls");
            sort_User_name(user);
            break;
        case '2':
            system("cls");
            sort_User_age(user);
            break;
        case '3':
            system("cls");
```

```
            sort_User_address(user);
            break;
         case '4':
            system("cls");
            sort_User_age(user);
            break;
         case '5':
            system("cls");
            break;
         default:
            system("cls");
            printf("\nInvalid choice & returning back ...\n");
            system("pause");
         }
         break;
      case '7':
         system("cls");
         exit(0);
         break;
      default:
         system("cls");
         printf("Invalid Choice");
         break;
      }
   }
   return 0;
}
```

Chapter 3

Performance Evaluation

3.1 OUTPUT

ENTER CHOICE

```
Enter Address: City
Enter Blance: 3000

Do you want to Add another user? (y/n): n

3 User added to file1

                        WELLCOME TO Fahim BANK MANAGEMENT SYSTEM
                                1. Add User Records
                                2. List User Records
                                3. Modify User Records
                                4. Delete User Records
                                5. Search User Records
                                6. Sort User Records
                                7. Exit System
  ______________________________________________________________________________
  ______________________________________________________________________________

 Your Choice:
```

INSERT

```
User 1 info:
Enter Name: Fahim
Enter age: 20
Enter Address: Banti
Enter Blance: 5000

Do you want to Add another user? (y/n): y

User 2 info:
Enter Name: Opu
Enter age: 20
Enter Address: Araihazar
Enter Blance: 2000

Do you want to Add another user? (y/n): y

User 3 info:
Enter Name: Alnami
Enter age:

20
Enter Address: City
Enter Blance: 3000

Do you want to Add another user? (y/n): n
```

List

```
Serial  Name                age     Address             Balance
0       Fahim               20      Banti               5000.00
1       Opu                 20      Araihazar           2000.00
2       Alnami              20      City                3000.00
Press any key to continue . . .
```

Modify

```
Serial  Name                age     Address             Balance
0       Fahim               20      Banti               5000.00
1       Opu                 20      Araihazar           2000.00
2       Sayem               20      Zidan               500.00
Press any key to continue . . .
```

SEARCH

```
Enter the name of the User you want to search: Fahim
Serial  Name                age     Address             Balance
0       Fahim               20      Banti               5000.00
Press any key to continue . . .
```

Sorting

```
Serial  Name                age     Address             Balance
0       Fahim               20      Banti               600.00
1       Opu                 20      Banti               400.00
Press any key to continue . . .
```

3.2 Results and Discussions

Analysis and Outcome

THIS BANKING SYSTEM WILL SERVE AS USEFUL APPROACH TO DEPOSIT AND WITHDRAW THE MONEY FOR THE PERSON.

IT REDUCES THE TIME TAKEN BY THE USER TO SAVE THE MONEY.

BANK SYSTEM DEVELOPED IS USER FRIENDLY.

IT REDUCES MANUAL WORK.

Chapter 4

Conclusion

The Bank Management System (BMS) is a web-based tool that allows the Bureau of the Fiscal Service to pay financial institutions for services rendered. BMS also has analytical tools that may be used to examine and approve pay, budgets, and outflows.

4.1 Scope of Future Work

For any system, present satisfaction is important, but it is also necessary to see and visualizes the future scope. Future enhancement is necessary for any system as the limitations that cannot be denied by anybody. These limitations can be overcome by better technologies. In my project, records of the customers and transactions are maintained. It will be

References?

[1] Used google to get the basic structure of my code.

[2] Got the idea for slideshare.com