| ***Computer Engineering Department*** |
| --- |
| ***CE100L: Computing Fundamentals & Programming*** |

| ***Course Instructor: Usama Bin Shakeel*** | ***Dated: 14/01/2022*** |
| --- | --- |
| ***Teaching Assistant: Aqsa Khalid*** | ***Semester: Fall 2021*** |
| ***Lab Engineer: Nadir Abbas*** | ***Batch: BSCE2021*** |

# **Lab 12B. Open Ended Problem Solving in C++**

| **Name** | **Roll number** | **Report**  **(out of 100)** | **Scaled to 10** | **Total**  **(out of 10)** |
| --- | --- | --- | --- | --- |
| NIMRA MAQBOOL | BSCE21012 |  |  |  |

Checked on: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## **Objective**

The objective of this lab is to understand open ended problem solving.

## **Equipment and Component**

| **Component Description** | **Value** | **Quantity** |
| --- | --- | --- |
| Computer | Available in lab | 1 |

## **Conduct of Lab**

1. Students are required to perform this experiment individually.
2. In case the lab experiment is not understood, the students are advised to seek help from the course instructor, lab engineers, assigned teaching assistants (TA) and lab attendants.

## **Theory and Background**

**Open-ended problem** is a problem that has several or many correct answers, and several ways to the correct answer(s). The Open-Ended Approach provides students with "experience in finding something new in the process"(Shimada 1997). It is basically facilitating the development of creative problem solving skills.

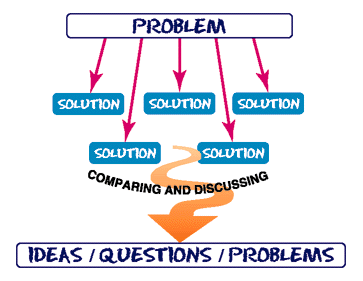


Figure 1: \*What is Open Ended Problem Solving??

\*https://mste.illinois.edu/users/aki/open\_ended/WhatIsOpen-ended.html

**Lab Task**

1. A company that wants to send data over the Internet has asked you to write a program that will encrypt it so that it may be transmitted more securely. All the data is transmitted as four-digit integers. Your program should read a four-digit integer in main() entered by the user and encrypt it as follows:

1. Replace each digit with the result of adding 7 to the digit and getting the remainder after dividing the new value by 10. Then swap the first digit with the third, and swap the second digit with the fourth. Then display the encrypted integer.

2. After encryption, the program will ask input an encrypted four-digit integer and decrypts it (by reversing the encryption scheme) to form the original number and return the decrypted number.

| void encryption(){  int arr[4]; //initializing  float digit1;  float digit2;  float digit3;  float digit4;  float temp1;  float temp2;  cout<<"please enter 4 positive integer ="; //asking to enter a positive integer  for(int i=0;i<4;i++){  cin>>arr[i]; //taking input from user in an array  }  cout<<endl;  for(int i=0;i<4;i++){ //displaying array of 4 integer  cout<<arr[i];  }  cout<<endl;  arr[0]=arr[0]+7; //adding 7 to the array number that is placed at 0 index  cout<<arr[0]; //displaying after adding 7 to it  arr[1]=arr[1]+7; //adding 7 to the array number that is placed at 1 index  cout<<arr[1]; //displaying after adding 7 to it  arr[2]=arr[2]+7; //adding 7 to the array number that is placed at 2 index  cout<<arr[2]; //displaying after adding 7 to it  arr[3]=arr[3]+7; //adding 7 to the array number that is placed at 3 index  cout<<arr[3]; //displaying after adding 7 to it  cout<<endl;  digit1=(arr[0]%10); //taking mode of 10 of the digit place at 0 index  cout<<"digit1 = "<<digit1<<endl; //displaying that digit  digit2=(arr[1]%10); //taking mode of 10 of the digit place at 1 index  cout<<"digit2 = "<<digit2<<endl; //displaying that digit  digit3=(arr[2]%10); //taking mode of 10 of the digit place at 2 index  cout<<"digit3 = "<<digit3<<endl; //displaying that digit  digit4=(arr[3]%10); //taking mode of 10 of the digit place at 3 index  cout<<"digit4 = "<<digit4<<endl; //displaying that digit  temp1=digit1; //storing the value of digit1 in temp1  digit1=digit3; //storing the value of digit3 in digit1  digit3=temp1; //storing the value of digit3 in temp1  temp2=digit2; //storing the value of digit2 in temp2  digit2=digit4; //storing the value of digit4 in digit2  digit4=temp2; //storing the value of digit3 in temp1  cout<<"the encrypted value is =" <<digit1<<digit2<<digit3<<digit4; //displaying the encrypted value  }  void decryption(){  int arr[4]; //initializing  float digit1;  float digit2;  float digit3;  float digit4;  float temp3;  float temp4;  cout<<"please enter 4 positive integer ="; //asking to enter a positive integer  for(int i=0;i<4;i++){ //taking input from user in an array  cin>>arr[i];  }  cout<<endl;  for(int i=0;i<4;i++){ //displaying array of 4 integer  cout<<arr[i];  }  cout<<endl;  arr[0]=arr[0]+3; //adding 3 to the array number that is placed at 0 index  cout<<arr[0]; //displaying after adding 3 to it  arr[1]=arr[1]+3; //adding 7 to the array number that is placed at 1 index  cout<<arr[1]; //displaying after adding 3 to it  arr[2]=arr[2]+3; //adding 7 to the array number that is placed at 2 index  cout<<arr[2]; //displaying after adding 3 to it  arr[3]=arr[3]+3; //adding 7 to the array number that is placed at 3 index  cout<<arr[3]; //displaying after adding 3 to it  cout<<endl;  digit1=(arr[0]%10); //taking mode of 10 of the digit place at 0 index  cout<<"digit1 = "<<digit1<<endl; //displaying that digit  digit2=(arr[1]%10); //taking mode of 10 of the digit place at 0 index  cout<<"digit2 = "<<digit2<<endl; //displaying that digit  digit3=(arr[2]%10); //taking mode of 10 of the digit place at 0 index  cout<<"digit3 = "<<digit3<<endl; //displaying that digit  digit4=(arr[3]%10); //taking mode of 10 of the digit place at 0 index  cout<<"digit4 = "<<digit4<<endl; //displaying that digit  temp3=digit1; //storing the value of digit1 in temp3  digit1=digit3; //storing the value of digit3 in digit1  digit3=temp3; //storing the value of digit3 in temp3  temp4=digit2; //storing the value of digit2 in temp4  digit2=digit4; //storing the value of digit4 in digit2  digit4=temp4; //storing the value of digit4 in temp4  cout<<"the decrypted value is =" <<digit1<<digit2<<digit3<<digit4; //displaying decrypted value  }  OUTPUT A;      OUTPUT B; |
| --- |

#### 2. A bank account charges $10 per month plus the following check fees for a commercial checking account:

#### $.10 each for fewer than 20 checks

#### $.08 each for 20-39 checks

#### $.06 each for 40-59 checks

#### $.04 each for 60 or more checks

#### The bank also charges an extra $15 if the balance of the account falls below $400(before any check fees are applied). Write a program that asks for the beginning balance and the number of checks written. Compute and display the bank’s service fees for the month and cheque amount will also be deducted from the balance

#### \*Input validation\*: Do not accept a negative value for the number of checks written. If a negative value is given for the beginning balance, display an urgent message indicating the account is overdrawn.

| void feeCharges(){  int beginningBalance; //initializing variables  int accountCharges=10; //declaring  int checksWritten = 0;  int moneyWithdrawn;  int totalMoneyTaken=0;  float check1=0;  float check2=0;  float check3=0;  float check4=0;  float serviceFee=0;  int moneyLeft;  cout<<"please enter beginning balance = "; //ask the user its beginning value  cin>>beginningBalance;  if(beginningBalance<0) { //condition to verify that value is not negative  cout << "your account is being overdrawn.";  }  else if(beginningBalance<400){ //condition to check that if the amount is less than 400  accountCharges = accountCharges + 15; //adding 15 in the 10 dollar which is constant  cout<<accountCharges;  }  cout<<"please enter how many checks have you written = "; //condition to check the writtenChecks  cin>>checksWritten;  if(checksWritten<0){ //condition to verify that the checksWritten is not negative  cout<<"u cant enter a negative number ."<<endl;  }  for(int i=0;i<checksWritten;i++){ //a loop for money user is with drawing from each check  cout<<"please enter the money u with drawn = ";  cin>>moneyWithdrawn;  moneyWithdrawn++;  totalMoneyTaken=totalMoneyTaken+moneyWithdrawn; //total money  }  cout<<"you have taken amount = "<<totalMoneyTaken<<endl; //displaying total money taken  if (totalMoneyTaken>beginningBalance){ //condition to verify that the money taken is less than the money that is already present in bank account  cout<<"you don't have enough money in your account, you can't withdraw it. "; //displaying  }  moneyLeft=beginningBalance-totalMoneyTaken; //to find the remaining money in bank account  cout<<"moneyLeft = "<<moneyLeft; //displaying moneyLeft in account  cout<<endl;  if(checksWritten<20){ //condition for the fee submit ion after every check that is less than 20  check1=(checksWritten\*(.01));  cout<<"check1 = "<<check1<<endl;  }  else if(checksWritten>=20 && checksWritten<=39){ //condition for the fee submit ion after every check that is less than equal to 39  check2=((.08)\*checksWritten);  cout<<"check2 = "<<check2<<endl;  }  else if(checksWritten>=40 && checksWritten<=59){ //condition for the fee submit ion after every check that is less than equal to 59  check3=((.06)\*checksWritten);  cout<<"check3 = "<<check3<<endl;  }  else { //condition for the fee submit ion after every check that is greater than 60  check4=((.04)\*checksWritten);  cout<<"check4 = "<<check4<<endl;  }  serviceFee=accountCharges+check1+check2+check3+check4; //adding all the quantities for total service fee  cout<<"The total service fee = "<<serviceFee<<endl; //displaying  }  OUTPUT |
| --- |

#### **Assessment Rubric for Lab**

**Method for assessment:**

Lab reports and instructor observation during lab sessions. Outcome assessed:

a. Ability to conduct experiments, as well as to analyze and interpret data (P) b. Ability to function on multi-disciplinary teams (A)

c. Ability to use the techniques, skills, and modern engineering tools necessary for engineering practice (P)

| Performance metric | Mapping (task no. and description) | | Max marks | Exceeds expectation | Meets expectation | Does not meet expectation | Obtained marks |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1. Realization of experiment (a) | 1 | Functionality | 40 | Executes without errors excellent user prompts, good use of symbols, spacing in output. Through testing has been completed (35-40) | Executes without errors, user prompts are understandable, minimum use of symbols or spacing in output. Some testing has been completed (20-34) | Does not execute due to syntax errors, runtime errors, user prompts are misleading or non-existent. No testing has been completed (0-19) |  |
| 2. Teamwork (b) | 1 | Group Performance | 5 | Actively engages and cooperates with other group member(s) in effective manner (4-5) | Cooperates with other group member(s) in a reasonable manner but conduct can be improved (2-3) | Distracts or discourages other group members from conducting the experiment (0-1) |  |
| 3. Conducting experiment (a, c) | 1 | On Spot Changes | 10 | Able to make changes (8-10) | Partially able to make changes (5-7) | Unable to make changes (0-4) |  |
| 2 | Viva | 10 | Answered all questions (8-10) | Few incorrect answers (5-7) | Unable to answer all questions (0-4) |  |
| 4. Laboratory safety and disciplinary rules (a) | 1 | Code commenting | 5 | Observes lab safety rules; handles the equipment and parts with care and adheres to the lab disciplinary guidelines aptly (4-5) | Generally observes safety rules and disciplinary guidelines with minor lapses (2-3) | Disregards lab safety and disciplinary rules (0-1) |  |
| 5. Data collection (c) | 1 | Code Structure | 5 | Excellent use of white space, creatively organized work, excellent use of variables and constants, correct identifiers for constants, No line-wrap (4-5) | Includes name, and assignment, white space makes the program fairly easy to read. Title, organized work, good use of variables (2-3) | Poor use of white space (indentation, blank lines) making code hard to read, disorganized and messy (0-1) |  |
| 6. Data analysis (a, c) | 1 | Algorithm | 20 | Solution is efficient, easy to understand, and maintain (15-20) | A logical solution that is easy to follow but it is not the most efficient (6-14) | A difficult and inefficient solution (0-5) |  |
| 7. Computer use (c) | 1 | Documentation | 5 | Timely documented (4-5) | Late documented (2-3) | Not documented (0-1) |  |
|  | Max Marks (total): | | 100 | Obtained Marks (total): | | |  |

Lab Engineer Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_