**CE100: Computer Fundamentals & Programming**

**Midterm Lab Exam (Fall 2021)**

***Name Muhammad Abubakar Saif*** ***Roll# BSCE21017***

**Instructions:**

1. Please try to solve all the questions in the allocated space.
2. Time allowed is the maximum time allowed to solve the paper.
3. It is a closed book and closed notes exam.
4. Every question has the marks mentioned in the right most column, so manage the time and answers according to mentioned numbers.
5. You should not be taking assumption without mentioning those.
6. **Anybody found cheating or helping any fellow during exam will get his/her paper cancelled immediately.**

***Time allowed: 150 Minutes Maximum Marks: 100***

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Marks Distribution** | | **Total** |
| **Question** | **1** | **2** |  |
| **CLO** | 2 | 2 |
| **Total Marks** | **75** | **25** | **100** |
| **Marks Obtained** |  |  |  |

**Teaching Team:**  Usama Bin Shakeel, Aqsa Khalid & Nadir Abbas



**Sets:** The collection of well-defined distinct objects is known as a set. The word well-defined refers to a specific property which makes it easy to identify whether the given object belongs to the set or not. The word ‘distinct’ means that the objects of a set must be all different.

Sets and arrays have several features in common. They both store a collection of values of the same type. A set is unordered and each element can only appear once in a set. While an array can contain duplicate elements, each value contained in a set is unique.

Example: arr below is an array which does not ensure that data is not duplicated therefore it is not a set, while arr2 is a set.

int arr [8] = {16, 2, 2, 2, 40, 77, 40, 12071};

int arr2 [5] = {16, 2, 77, 40, 12071};

|  |  |  |
| --- | --- | --- |
| Q1 | For Question 1, all the functions should be created in q1.h and q1.cpp files.   * Write a menu function that gives following options:  \* Press 1 for creating new array of size N \* Press 2 to check if current array is a set  \* Press 3 to add new element to array ensuring that it remains a set \* Press 4 to display array (Recursion) \* Press 5 to extract set from an array * Explanation for 1:   Take the size from user, and create an array of that size, all data values should be set to -1, to show that user haven’t provided the data yet.   * Explanation for 2:   Check only the elements that have been added by user, to verify if the properties of set are being maintained or not.   * Explanation for 3:  Check whether a new element can be added to the array? If so verify if the property of set would be maintained? then only add it to the array. * Explanation for 4:  Using recursion, display the elements of the array that have been filled by the user. * Explanation for 5: if an array is sent to this function, it verifies if it is already a set, if not, we will create a new array, copy data while maintaining set property of uniqueness, and return that array. | **75**  (15 + 12 x 5) |
|  | #include <iostream>  #include "q1.h"  using namespace std;  int size;  int\* user\_arr = new int[size];  void displayMenu() {  bool validation;  int user; //used for the purpose of asking choice from user  cout << "\* Press 1 for creating new array of size N" << endl;  cout << "\* Press 2 to check if current array is a set " << endl;  cout << "\* Press 3 to add new element to array ensuring that it remains a set " << endl;  cout << "\* Press 4 to display array (Recursion) " << endl;  cout << "\* Press 5 to extract set from an array " << endl;  cin >> user;  switch (user)  {  case 1:  cout << "Enter size of array you want to create: ";  cin >> size;  arrayCreation();  break;  case 2:  validation = setValidator();  if (validation == true) {  cout << "Current array is the set." << endl;  } else cout << "Current array is not the set." << endl;  break;  case 3:  arrayEditor();  break;  case 4:  cout << "Elements of Array: " << endl;  displayArray(0);  break;  case 5:  extractionSet();  break;  default:  break;  }  }  void arrayCreation() {  int option;  for (int i = 0; i < size; i++) {  user\_arr[i] = -1;  }  cout << "Array created of given size." << endl;  cout << "Do you want to fill the array: Press 1 for Yes else Press 0" << endl;  cin >> option;  if (option == 1) {  for (int i = 0; i < size; i++) {  cout << "Enter the element#" << i + 1 << " of array: (if you want to skip this element, enter -1) " << endl;  cin>>user\_arr[i];  }  }else return;  }  bool setValidator() {  for (int i = 0; i < size; i++) {  if (user\_arr[i] == user\_arr[i + 1] && user\_arr[i] != -1) {  return false;  }else return true;  }  }  void arrayEditor() {  int runCheck=0;  bool check;  for (int i = 0; i < size; i++) {  if (user\_arr[i] == -1) {  cout << "Enter the new element you want to add: ";  cin >> user\_arr[i];  check = setValidator();  if (check == false) {  cout << "You can't add this number in array, it violates the basic rules of Set theory." << endl;  user\_arr[i] = -1;  }  runCheck++;  }  }  if (runCheck == 0) {  cout << "You can't add any element in array, it is already full or may not exist." << endl;  }  return;  }  void displayArray(int i) {  if (i == size) {  return;  }  else {  if (user\_arr[i] != -1) {  cout << "Element#" << i+1 << " :" << user\_arr[i] << endl;  }else cout << "Element#" << i+1 << " :" << "Empty" << endl;  i++;  }  displayArray(i);  }  void extractionSet() {  int j = 0;  bool valid = setValidator();  if (valid == false) {  int\* copyArr = new int[size];  for (int i = 0; i < size; i++) {  if (user\_arr[i - 1] == user\_arr[i]) {  copyArr[j] = user\_arr[i];  j++;  }  }  for (int i = 0; i < size; i++) {  user\_arr[i] = -1;  user\_arr[i] = copyArr[i];  }  delete[] copyArr;  }  } |  |

|  |  |  |
| --- | --- | --- |
| Q2 | For Question 2, all the functions should be created in q2.h and q2.cpp files.  Write a program to help a university system to store records for its employees. You have to perform the following task:   1. Define a struct **facultyMember** with the following attributes:  * ID number (int) * First Name (string) * Last Name (string) * Designation (string) e.g. Assistant professor, Lecturer etc.  1. Write a method to take input from user and write it to file in append mode | **25   (10+ 15)** |
|  | struct facultyMember {  int ID;  string firstName;  string lastName;  string designation;  };  void universityRecord() {  facultyMember store;  fstream file;  file.open("University Record.txt", ios::app);  cout << "Enter the Employee ID: ";  cin >> store.ID;  file << store.ID << " ";  cout << "Enter the Employee first name: ";  cin >> store.firstName;  file << store.firstName<<" ";  cout << "Enter the Employee Last Name: ";  cin >> store.lastName;  file << store.lastName << " ";  cin.ignore();  cout << "Enter the Employee Designation: ";  getline(cin, store.designation);  file << store.designation << " ";  file << endl;  } |  |