Solution

BAHRIA UNIVERSITY

ISLAMABAD CAMPUS

Department of Computer Science

Finalterm Examination

BSIT 1 [A, B]/ BSCS 0[S, T]

(Spring 2022 Semester)

Paper Type: Descriptive

Course:	**Computer Programming**	Date: 18-07-2022
Course Code:	CSC-113	Time: Session II
Faculty's Name:	**Mr. Abrar Ahmed/Ms. Nabia Khalid/ Ms. Maryam Aslam**	Max Marks: 50
Time Allowed:	150 minutes	Total Pages: 10

STUDENT'S NAME (IN FULL): _____

ENROLMENT NO: _____CLASS/Section_____

1. Write your full name and other particulars clearly and legibly. Write on both side of the papers. No page to be torn and taken out of examination venue.
2. Read the instructions on question paper and answer book carefully and understand.
3. Paper will commence at exact time. Be punctual and be inside the examination hall at least 15 minutes before paper start time.
4. Be seated as per seating plan depicted in the Examination Admit Slip.
5. Students after start of paper will not be permitted to go to washrooms/toilets or any other place outside the examination venue.

***N. B:** read carefully the instructions given overleaf*

HALL NO: _____ INVIGILATOR'S SIGN: _____

INVIGILATOR'S NAME: _____

Instructions about the Paper

I. There are total SIX questions. All questions are compulsory.
II. The paper is closed book.
III. The students are not allowed any helping material (books, tables, formulas, etc).
IV. Use blue, black or blue-black ink only. Do NOT use lead pencil especially.
V. Do not cheat.

Question#	Total Marks	Obtained Marks	Question#	Total Marks	Obtained Marks
Question-1	15		Question-5	9	
Question-2	5		Question-6	8	
Question-3	5				
Question-4	8				
Total/50			Signatures		

Enrollment Number: _____

INSTRUCTIONS FOR STUDENTS

1. No student would be allowed to sit in the Examination venue without showing Examination Admit Slip to the invigilator. No student should allow anyone to impersonate him/her. This may result in serious consequences even to the extent of cancellation of registration from Bahria University.

2. Students prior entering the Examination venue should ensure that they are not in possession of written material of any sort. All such material is to be left outside the venue Examination. Any written material found possession of a student, whether that material is related or unrelated to the paper, will result in grade 'F' in the relevant paper.

3. Writing on palm, arm or anywhere on the candidate's body/clothing is considered enough proof of cheating, which will result in award of grade 'F' in the paper.

4. Any attempt to copy/take or give help in examination is an offence, punishable even to the extent of expulsion from the Institution.

5. Books and notes are not to be brought inside the examination hall except in case of open book exam.

6. Bring your own pen, pencils erasers, scales and calculators. Borrowing at the places of examination is not permitted. Special/programmable calculators (except where permitted), electronic notebooks, mobiles phones, PDAs and any other electronic accessories are prohibited at the examination venues.

7. All rough work is to be done on right side of the answer book, opposite the same question.

8. Additional sheets or graph sheet etc, if used, are to be properly tagged. Serial number of extra book(s) taken (if any) should be entered in the specified box on the main Answer book.

9. Do not ask for any help form the invigilators in solving questions. This may be taken as an act of academic dishonestly and dealt accordingly. You may seek invigilator's assistance regarding misprinting. How and what to write should not be asked. Any query related to the question paper is to be clarified by concerned faculty member within first thirty minutes of the paper only. The query is to be addressed to all the students loudly by the concerned faculty.

10. Possession of firearms, knives etc, inside and in the vicinity of Examination Half is a punishable crime under the country's law.

11. Disrupting the Examination venue by shouting or by rowdy acts, will be considered as serious punishable act under the country's law.

12. You are required to be respectful and polite and polite towards the invigilation and admin staff. Show of temper, anger, misbehavior, misconduct or disrespectful utterances will be dealt with serious punishment.

13. Eatables, beverages and smoking is not permitted at Examination venue.

Certified that I have read and understood the instruction for compliance in the Examination hall/venue and I hereby undertake to abide by these in their true letter and spirit. I also declare on oath/affirm that I shall not challenge any penalty imposed on me by the Competent Authority for violating any of the instruction.

Signature of the Student: _____

1) Write the output of the following codes; (Marks 3 + 3 + 3 + 2 + 2 + 2 = 15)

a)

```
#include <iostream>
using namespace std;
struct Box
{
    int length, breadth, height;
};
int main()
{
   Box B1 = {50, 55, 25}, B2, B3;
   ++B1.height;
   B3=B1;
   ++B3.length;
   B3.breadth++;
   B2=B3;
   B2.height +=5;
   B2.length--;
   cout << B1.length << "  "<<B2.breadth<<"  "<<
B3.height<<endl;
return 0;
}
```

Output

50 56 26

b)

```
int main()
{
      int x;
      int y;

      int *xptr = &x;
      int *yptr = &y;
      x = 50;
      y = 70;
      xptr = yptr;
      cout << endl;
      *xptr = 100;
      cout << x << " " << y<<endl;
      cout << *xptr << " " << *yptr << endl;

      _getch();
      return 0;
}
```

50 100

100 100

c)

```
int main()
{
	int x, * p, * q;
	int arr[3] = { 0 };
	p = arr;
	q = p;
	*p = 4;
	for (int j = 0; j < 2; j++)
	{
		x = *p;
		p++;
		*p = (x + j);
	}
	for (int k =0; k<3; k++)
	{
		cout << *q<< " ";
		q++;
	}
	_getch();
	return 0;
}
```

4 4 5

d)

```
for (int loop = 1; loop <= 10; loop++)
	{
		if (loop % 3 == 0)
			continue;
		cout << loop << "    ";
	}
```

1 2 4 5 7 8 10

e)

```
	char myChar = 'A';
	if (myChar >= 65 && myChar <= 122)
	{
		myChar += 32;
		cout << myChar++ << endl;
		cout << myChar << endl;
	}
```

a

b

f)

```
	int x = 3;
	while (x != 7)
	{
		cout << x << endl;
		x += 2;
	}
```

3

5

2) Write a function that takes a positive integer from the user and declares one dimensional dynamic array using new keyword. Function then initializes the array with random 2-digit values and prints the sum of even values and sum of odd values separately. (Marks = 5)

```
void DynamicArray()
{
    srand(time(0));

    int size;
    cout << "Enter Size of Array" << endl;
    cin >> size;

    int* array = new int[size];

    cout << "Array Elements are" << endl;

    for (int i=0; i< size; i++)
    {
        array[i] = 10 + rand() % 90;
        cout << array[i] << "  ";
    }

    cout << endl;

    int evensum = 0, oddsum=0;

    for (int i = 0; i < size; i++)
    {
        if (array[i] % 2 == 0)
            evensum += array[i];

        else
            oddsum += array[i];
    }
    cout <<"Sum of even values are" <<"  "<<evensum <<endl;
    cout << "Sum of odd values are" << "  " << oddsum << endl;
}
```

3) Write a program that will take an array and its size from user and display average of array elements. Create a function named GetAverage(), function will takes an array and its size as arguments and will return average of array elements to main function. (Marks = 5)

```
#include<iostream>
#include<conio.h>
#include <time.h>
#include <fstream>
#include <string>

using namespace std;

double GetAverage(int arr2[], int size)
{
	int sum = 0;

	for (int i = 0; i < size; i++)
	{
		sum += arr2[i];
	}

	double avg = (double) sum / size;
	return avg;
}

int main()
{

	int size;
	cout << "Enter array size"<<endl;
	cin >> size;

	int* arr1 = new int[size];

	cout << "Enter array elements" << endl;
	for (int i = 0; i < size; i++)
	{
		cin >> arr1[i];
	}

	cout << "Average of Array Elements are "<<GetAverage(arr1, size);

	_getch();
	return 0;
}
```

4) Write the functions vowelsCount() and wordsCount() separately, both takes a char [] (character array) as argument containing a sentence and display the total number of vowels and the total number of words in that sentence accordingly. (Marks = 4 + 4)

```
void printVowelCount(char myString[])
{
	int size = strlen(myString);

	int vowelCounter = 0;

	for (int index = 0; index < size; index++)
	{
		if (myString[index] == 'a' || myString[index] == 'e' ||
myString[index] == 'i' || myString[index] == 'o' || myString[index] == 'u')
			vowelCounter++;
	}

	cout << "Total Vowels in this string are : " << vowelCounter << endl;
}


// ************************************************************************


void printWordsCount(char myString[])
{
	int size = strlen(myString);

	int wordCount = 1;

	for (int index = 0; index < size; index++)
	{
		if (myString[index] == ' ' || myString[index] == '\n')
		{
			cout << endl;
			wordCount++;
		}
		else
			cout << myString[index];
	}

	cout << endl << "Total words in string are : " << wordCount << endl;
}
```

5) **Write a program that read quizzes marks and assignments marks from two different files "quiz.txt" and "assignment.txt" respectively. Calculate the internal marks (i.e. quizzes marks + assignment marks) and store the internal marks into new file named "internal.txt". (Marks = 3+3+3)**

```
#include<iostream>
#include<conio.h>
#include <fstream>
#include <string>
using namespace std;
int main()
{
	ifstream assign;
	assign.open("assignment.txt");
	if (!assign)
	{
		cout << "file is not opened";
		exit(0);
	}
	int a;
	while (!assign.eof())    //  while (assign>>a)
	{
		assign >> a;
		cout << a;
	}
	assign.close();
//************************************************************
	ifstream quiz;
	quiz.open("q.txt");

	int b;

	while (!quiz.eof())      //  while (quiz>>b)
	{
		quiz >> b;
		cout << b;
	}
	quiz.close();
//************************************************************
	int c = a + b;

	ofstream internal;
	internal.open("internal.txt");

	int i = 1;
	while (i<2)
	{
		internal << c;
		cout << c;
		i++;
	}
	internal.close();
	_getch();
	return 0;
}
```

6) **Question:** **(Marks = 2+1+2+1+2)**

a) **Define a struct *houseType* to store the following data**
   i. Style, numOfBedrooms, numOfBathrooms, numOfCarsGarage, yearBuilt, finishedSquareFootage, price and tax.

b) **Declare variables *oldHouse* and *newHouse* of type *houseType***

c) **Store the following information into *oldHouse*: Style—Two-story, number of bedrooms—5, number of bathrooms—3, number of cars garage—4, year built—1975, finished square footage—3500, price—675000, and tax = 12500**

d) **Copy the values of the components of *oldHouse* into the corresponding components of *newHouse*.**

e) **Display the information stored, by creating a function *displayInfomation* () by using variable *newhouse*.**

```
#include<iostream>
#include<conio.h>
#include <fstream>
#include <string>
using namespace std;

struct houseType
{
	string Style;
	int noOfBedrooms;
	int noOfBathrooms;
	int numOfCarsGarage;
	int yearBuilt;
	double finishedSquareFootage;
	double price;
	double tax;
};

void displayIn(houseType newHouse1);

int main()
{
	houseType oldHouse, newHouse;

	oldHouse.Style = "Two-story";
	oldHouse.noOfBedrooms = 5;
	oldHouse.noOfBathrooms = 3;
	oldHouse.numOfCarsGarage = 4;
	oldHouse.yearBuilt = 1975;
	oldHouse.finishedSquareFootage = 3500.0;
	oldHouse.price = 675000.0;
	oldHouse.tax = 12500.0;

	newHouse = oldHouse;

	displayIn(newHouse);

	_getch();
	return 0;
}
void displayIn(houseType newHouse1)
{
	cout << "Style of House is" << "  " << newHouse1.Style << endl;
	cout << "No of Bedrooms are" << "  " << newHouse1.noOfBedrooms << endl;
	cout << "No of Bathrooms are" << "  " << newHouse1.noOfBathrooms << endl;
	cout << "No of Car Gerages are" << "  " << newHouse1.numOfCarsGarage << endl;
	cout << "Year of Built" << "  " << newHouse1.yearBuilt << endl;
	cout << "Total Finised Square feets are" << "  " << newHouse1.finishedSquareFootage << endl;
	cout << "Price of the House is" << "  " << newHouse1.price << endl;
	cout << "Total tax of the house is" << "  " << newHouse1.tax << endl;
}
```

Good Luck