CS-1002: Programming Fundamentals

Serial No:

Sessional Exam-I

Total Time: 1 Hour

Total Marks: 50

Monday, 14th March, 2022

Course Instructors

Mudassar Aslam, Uzma Bibi, Nimra Shahid

_____

Signature of Invigilator

_____ _____

Student Name Roll No. Section Signature

DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.

Instructions:

1. Attempt on question paper. Attempt all of them. Read the question carefully, understand the question, and then attempt it.
2. Time for **Question 1** is **15 minutes.** Detach the last page (**page number 09**) which is Answer Page for Question 1. You must attempt Question 1 on page number 09 and **return after 15 minutes**. Don't forget to write your name and roll number on the Question 1 answer sheet as well before you start.
3. No additional sheet will be provided for rough work. Use the back of the last page for rough work.
4. After asked to commence the exam, please verify that you have **nine (9)** different printed pages including this title page. There are a total of **3** questions.
5. Calculator sharing is strictly prohibited.
6. Use permanent ink pens only. Any part done using soft pencil will not be marked and cannot be claimed for rechecking.

	Q-1	Q-2	Q-3	Total
Marks Obtained				
Total Marks	**20**	**20**	**10**	**50**

Question 1 [20 Marks]
To be answered on the last page, detach and return in 15 minutes

1. Signed, unsigned, long and short are some of the ___
a) Void
b) Data
c) Derived data
d) Modifiers

2. Choose the correct order of operators in terms of lowest to highest precedence
a) Arithmetic, logical, relational
b) Logical, relational, arithmetic
c) Arithmetic, relational, logical
d) Relational, arithmetic, logical

3. Variable names must begin with ___
a) #
b) $
c) Number
d) Letter

4. Conditional operator (?:) is a handy operator which acts as a shortcut for ____
a) if-else statement
b) switch statement
c) break statement
d) goto statement

5. The class istream is derived from _____ which contains all the necessary functions for handling input
a) Base class
b) ios class
c) derived class
d) both a and c

6. ____ is an identifier that can be inserted into an output stream or extracted from an input stream in order to produce the desired effect.
a) Stream
b) Manipulator
c) this
d) Flag

7. Variable1 = expression; which one is evaluated first?
a) Variable1
b) Expression
c) Both are evaluated equally
d) None of them

8. Empty string is denoted by
a) “0”
b) “ “
c) 0;
d) 0

9. The return 0 statement at the end of the program indicates what?
a) That the program terminated their execution
b) That the control return to the operating system
c) 0 indicates that program ended successfully
d) All of them

10. Which of the following C++ operator associativity starts from right side?
a) Add operator +
b) Simple assignment operator =
c) Subtract operator –
d) Bit shift operator <<

11. Which of the following escape sequence represents the carriage return?
a) \r
b) \n
c) \n \r
d) \t

12. Which of the following is the correct syntax to add the header file in the C++ program?
a) #include<userdefined>
b) #include "userdefined.h"
c) <include> "userdefined.h"
d) Both A and B

13. The most efficient data type for a variable that stores the number 4.6e20 is the _____ data type
a) Character
b) Double
c) Float
d) Long integer

14. The bitwise OR operator is a
a) Unary operator
b) Binary operator
c) Ternary operator
d) None of them

15. If p and q are assigned the values 2 and 3 respectively then the statement p = q++
a) gives an error message
b) assigns a value 4 to p
c) assigns a value 3 to p
d) assigns a value 5 to p

16. Assuming the following three operators appear in an expression (without parentheses), which of the operators will be performed first?

a) &&
b) !
c) ||
d) None of the above

17. Minimum number of temporary variable(s) needed to swap the contents of 2 variables is/are:

a) 1
b) 2
c) 3
d) 0

```
x = x + y
y= x - y
x =x - y
```

18. Using keywords for variable names will result in a _____

a) Runtime error
b) Compile error
c) Syntax error
d) Semantic error

19. The extraction operator when applied to a character ignores whitespace.

a) True
b) False

20. float 5/2 results to

a) 2
b) 2.5
c) Error
d) Garbage value

-- Answers of this section MUST BE RETRUNED in the first 15 minutes on Page # 09 BEFORE starting the next question –

Question 2 [20 Marks]

Check the code snippets given below and identify syntax errors if any. If you find an error, fix it and then write the output of the code. Assume that all required header files are included and this code is written in the main() function.

int x = 8, y =5, z=1, a=15 ; bool hello_1= x > y > z ; bool hello_2= a=y=6 ; bool hello_3 = x >= 0 && y >=5 && z ; cout<< hello_1 << ":"<< hello_2<< ":"<< hello_3<<endl; cout<< a << ":"<< x<< ":"<< y<< ":"<< z;	Relational operator: L to R Hello_1 = 0 ((8>5)>1) Hello_2 = 1 Hello_3 = 1 (1)&&(1)&&(1) 0:1:1 6:8:6:1
int x, y=2; x = 1;	x = 1 , y = 2 y = 2*2 = 4

y = ++x * y;
cout << x <<" :"<< y <<endl;
x += 5;
y = (x+=1) * (x+1);
cout << x <<" :"<< y;
	2 : 4
	x = 5 +2 = 7
	y = (8) * (9)
	8 : 72

char alphabet= 'N';
int i = 7;
i = 'D'- alphabet + 5;
cout << static_cast(char)(alphabet+3) << endl ;
cout << i << endl;
cout << char(i+ alphabet) << endl;
	Syntax error
	cout << static_cast<char>(alphabet+3) << endl ;
	
	Ascii of A= 65, D = 68, N = 78
	i = 68 – 78 + 5 = -5
	Q (78 + 3 = 81)
	-5
	I (-5 + 78 = 73)

int a = 0, b=36;
float f=3.9;
b+=(a = 50)*(int)f%3+5.1-6.8*10-b/5;
cout << a << "$" << b;
	b = b + 50 * 0 +5.1 – 68 – 7
	b = 41.1-78 = -33
	
	50$-33

int i = 100, j = 30;
if(i > j);
i++;
else i--;
cout<< "value of \"i\\:"<< i;
	Syntax error because
	if (i > j);
	by removing semicolon
	value of i: 101

const int U = 8, L = 2;
int num1, num2, num3 = 12, num4 = 3;
num1 = num3 > num4 ? num4 > U ? num3 : L : U;
num2 = num3 < num4 ? num4 > U ? num3 : L : U;
cout << num1 << endl;
cout << num2 << endl;
	Num1 = 12 > 3 True
	Num1 = 3 > 8 False
	Num1 = 2
	Num2 = 12 < 3 False
	Num2 = 8

Let's suppose we have a special data type called **my_int** which takes 5 bytes in the memory. We declare following variables of this data type. unsinged my_int a; my_int b;	Maximum number that can be stored in a? 1,099,511,627,775 Minimum number that can be stored in a? 0 Maximum number that can be stored in b? 549,755,813,887 Minimum number that can be stored in b?-549,755,813,888 5bytes = 40 bits **Unsigned:**

	`int max = pow(2, number of bits assigned to data type) – 1;`
	max = $(2^{40} - 1)$ = 1,099,511,627,775
	
	Signed:
	
	`int min = (pow(2, number of bits assigned to data type) / 2) * -1;`
	
	min = -1* $(2^{40} / 2)$ = - 549,755,813,888
	
	`int max = (pow(2, number of bits assigned to data type) / 2) – 1;`
	
	min = $(2^{40} / 2)$ - 1 = 549,755,813,887

cout << setw(10) << setprecision(3) << setfill(*); cout << 34.267 << endl;	Syntax error cout << setw(10) << setprecision(3) << setfill('*'); ******34.3

string str;
cin>>str;
//user enters -> Capital City
cout<<str;
cin>>str;
//user enters -> surprise
cout<<str;
	CapitalCity
	
	Use the ignore() Function to **Discard Unwanted Command Line User Input**.
	The ignore() function is a member function of std::basic_istream and is inherited by different input stream classes. The function discards the characters in the stream until the given delimiter, inclusive, and then extracts the stream's remainder.
	
	`string str;`
	`cin >> str;`
	`//user enters -> Capital City`
	`cout << str;`

	cin.ignore(numeric_limits<std::streamsize>::max(), '\n'); cin >> str; //user enters -> surprise cout << str;

int a = INT_MAX;
cout<<a++<<endl;
cout<<a--<<endl;
cout<<a<<endl;
	2147483647
	-2147483648
	2147483647

Question 3 [10 Marks]

Write a program to get a number from user and check if that integer is a power of 2. Use bitwise operators to check (see some facts given below for the hint). Input validation is must: number should not be zero, use of any power function is not allowed (10 marks).

1. If number is power of 2 then then find all roots of a quadratic equation using the nested switch statement. Take a, b, c as input from the user and display the both roots. Format your root values in fixed-point notation, with two decimal places of precision.
2. If it is not power of 2, then display a message that the number is not power of 2.

Note: You may use the sqrt() function of <math.h> library to find the square root.

Power of 2: Some facts about numbers which are power of 2. You may use these facts to check if it is power of 2 or not. Suppose N is power of 2, then:

- all bits of N are zero except (any) one bit.
- All least significant bits of N-1 before the ON bit of N are always ON.

Quadratic equation: In elementary algebra quadratic equation is an equation in the form of

$$ax^2 + bx + c = 0$$

Solving quadratic equation

A quadratic equation can have either one or two distinct real or complex roots depending upon the nature of the discriminant of the equation. Where discriminant of the quadratic equation is given by

$$\Delta = b^2 - 4ac$$

Depending upon the nature of the discriminant, formula for finding roots can be given as:

- Case 1: If the discriminant **is positive**. Then there are two real distinct roots given by.

$$\frac{-b+\sqrt{\Delta}}{2a} \quad \text{and} \quad \frac{-b-\sqrt{\Delta}}{2a}$$

- Case 2: If **discriminant is zero**. Then it has exactly one real root given by.

$$-\frac{b}{2a}$$

- Case 3: If **discriminant is negative**. Then it will have two distinct complex roots given by.

  root 1 = root 2 $=\frac{-b}{2a}$, Imaginary = $\frac{\sqrt{-\Delta}}{2a}$

Solution:

```
int main()
{
bool flag = 0;
   int num;

   cout << "Enter the number ";
   cin >> num;

   flag = !((num) && (num & (num - 1))); // check the power of 2
   if (flag == 1)
   {
```

```
        cout << "Number is a power of 2 \n";

        float a, b, c;
        float root1, root2, imaginary;
        float discriminant;

        cout << "Enter values of a, b, c of quadratic equation (aX^2 + bX + c): ";
        cin >> a >> b >> c;

        /* Calculate discriminant */
        discriminant = (b * b) - (4 * a * c);


        /* Compute roots of quadratic equation based on the nature of discriminant */
        switch (discriminant > 0)
        {
        case 1:
           /* If discriminant is positive */
           root1 = (-b + sqrt(discriminant)) / (2 * a);
           root2 = (-b - sqrt(discriminant)) / (2 * a);

           cout << "Two distinct real roots exists: " << setprecision(2) <<fixed << root1 << root2;
           break;

        case 0:
           /* If discriminant is not positive */
           switch (discriminant < 0)
           {
           case 1:
              /* If discriminant is negative */
              root1 = root2 = -b / (2 * a);
              imaginary = sqrt(-discriminant) / (2 * a);

              cout << "Two distinct complex roots exists: "  << setprecision(2) << fixed << root1 << imaginary << root2 <<
imaginary;
              break;

           case 0:
              /* If discriminant is zero */
              root1 = root2 = -b / (2 * a);

              cout << "Two equal and real roots exists: " << setprecision(2) << fixed <<  root1 << root2;

              break;
           }
        }
     }
     else
     {
        cout << "Number is not a power of 2 \n";
     }
```