

SHRI T.P. BHATIA COLLEGE OF SCIENCE

CLASS: XI

FIRST SEMESTER EXAMINATION 2016-17

MARKS: 50

DATE: 18/10/2016

COMPUTER SCIENCE - I

DURATION: 3 HRS

Q1.A. Select the most appropriate option and rewrite the following statements. [4]

- _____ is not true if clrscr() is excluded in C++ program.
a) Compiler would give an error
b) No error, but output won't be displayed.
c) It clears the output screen
d) both a) and b)
- _____ data-type needs 2-byte of memory space.
a) int
b) unsigned
c) short
d) All of these
- A C++ program must be saved with _____ extension.
a) C
b) C++
c) CPP
d) CP
- _____ is a valid operator in C++.
a) ;
b) +
c) ,
d) All of these

Q1.B. State the error(s) in the following C++ statements, **if any**. Also, write the output for the statement **that doesn't show any error**. Consider the initial values: int a = 50, c, b; float d = 3.0; Use the changed values of these variables, if any, for the subsequent statements. [6]

- c = a % d;
Error: % (modulus operator) **cannot** work on float value and variable **d** is of type float.
- cout << a++;
50 [Execution: ++ is post-fixed so first value of **a** is printed as 50 then a will be incremented to 51]
- d/d * -a = b;
Error: The left-operand to the assignment operator cannot be an expression; it must always be a variable.
- c = !(a != a);
c = 1 [Execution: !(a != a) means !(51 != 51) = !(0) = 1, hence 1 is assigned to c]
- b = 30000 + d;
b = 30003
[Execution: 30000.0 + 3.0 is 30003.0 but left operand is **b** of type int hence only integral part will be stored]
- cout << ++50;
Error: ++ cannot work on constant value.

Q2.A. Attempt any three of the following. [9]

- Define a variable. Mention the rules for naming a variable.
Ans: A variable is a name given to a single or multiple memory location(s) that store value(s).
Rules for naming a variable:

- a) A variable name can have alphabets, digits or/and an underscore.
 - b) The length of the name should not be more than 32 characters.
 - c) It must not start with a digit. Although, the starting character can be an alphabet or an underscore.
 - d) Existing keywords' names must not be used for variables.
2. Write an algorithm that reads x and y coordinates of a point from the user. Check whether this point falls on the origin of the coordinate plane or not. If not, then check in which quadrant does it lie in the plane.
1. PRINT "Enter x and y coordinates of a point"
 2. READ x, y
 3. IF (x = 0 AND y = 0) THEN
 PRINT "The point lies on origin"
 - ELSE IF (x > 0 AND y > 0) THEN
 PRINT "The point lies in first quadrant"
 - ELSE IF (x < 0 AND y > 0) THEN
 PRINT "The point lies in second quadrant"
 - ELSE IF (x < 0 AND y < 0) THEN
 PRINT "The point lies in third quadrant"
 - ELSE IF (x > 0 AND y < 0) THEN
 PRINT "The point lies in fourth quadrant"
 - ELSE
 PRINT "The point doesn't lie in any of the quadrants neither on origin"
4. END

3. How is **switch-case** different from **if-else**?

Ans: 1) A float value cannot be checked in switch-case.

2) A break statement is needed to end the switch block unlike if-else. After the execution of, **if** block the control automatically skips the else block; no exclusive statement is required to end an **if** or **else** block.

3) Cases can never have a variable expression but if-else can! **[Example: case a + 2: is an invalid statement.]**

4) Cases can never have a relational or logical expression **[Example case a>2:]** unlike if-else.

4. Mention any two keywords in C++. Also, explain them.

Ans: 1) **int**: It's a data-type in C++, used to declare a variable. It allows a variable to store an integer value within range of -32678 to +32767 with a memory space of 2 bytes.

2) **break**: It is a keyword that forcefully takes the control out of the block, skipping all the following statements within the block, where it appears; without checking any condition unlike **if**

statement. Hence, it is called an unconditional branching statement. But, there are some limitations to it, which is it can be used only within switch-case, and loop blocks.

Q2.B. Write a single C++ statement that stores the square root of x in y. [1]

$y = \text{sqrt}(x);$

Q3. Attempt any five of the following. [10]

1. Differentiate between '=' and '==' operators. **[Minimum 2 distinct points]**

=	==
It assigns the value of its right-operand to its left operand.	It checks whether the left operand is equal to the right operand or not. If true then returns 1 else returns 0 as false value.
The left operand must be variable. Although, the right-operand can be any expression evaluating to a value.	Both the operands can be any expression evaluating to a value.
Example: a = 2;	Example: a==b+c

2. In the statement **#include<math.h>**, what does 'h' stand for? Why is this statement used in a C++ program?

ANS: The 'h' here denotes the extension of a header file. The Pre-defined functions to perform mathematical tasks are collectively defined in "math.h" file. To make use of such functions in any C++ program, we need to import this file to make the compiler aware about its location (where it has been defined).

3. Write a single print statement in C++ to give the following output.

CLASS: XII EXAM SEAT NO.: A123456
 DIV : Y NO. OF EXPERIMENTS PERFORMED: 16

cout<<"CLASS: XII\tEXAM SEAT NO.: A123456\nDIV : Y\tNO. OF EXPERIMENTS PERFORMED: 16";

4. Write a C++ arithmetic statement for the mathematical expression

$$\frac{2v + 6.22(c + d)}{g + v} = R$$

R = (2 * v + 6.22 * (c + d)) / (g + v);

5. Justify the statement: The operator '-' in C++, is a unary and binary operator as well.

In C++, there are many symbols/operators used for various purposes. One of them is, the '-' operator. It is termed as unary operator when used with single operand otherwise it acts as binary operator when used with two operands. Well, in both cases it performs different tasks. In case of one operand, it changes the sign of the operand and in later one it performs subtraction. **Example: cout<<-a; cout<<a-b;**

6. If two operators of same precedence are used in an expression then which operator's task will be performed first and why? Give an example.

If the operators with same precedence are used in an expression then the compiler resolves the ambiguity with the operations using 'associativity'. Each set of operators have been given associativity as left to right or right to left. If the associativity is left to right then whichever operator's expression appears first will be evaluated followed by the next. Example, $a * b * c$. $a * b$ will be solved first and the result will be multiplied with c .

Q4A. Explain the following C++ terms, with an example.

[6]

1. `main()`: The execution of every C++ program starts from `main()`. So, `main()` is a must function in any C++ program. The instructions for whatever task(s) is/are to be performed must be enclosed within a pair of braces of `main()` block.

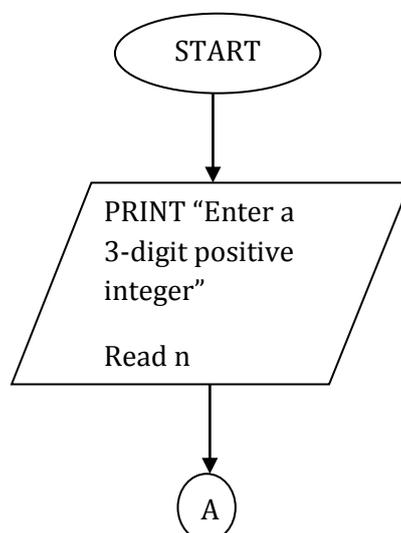
Example:

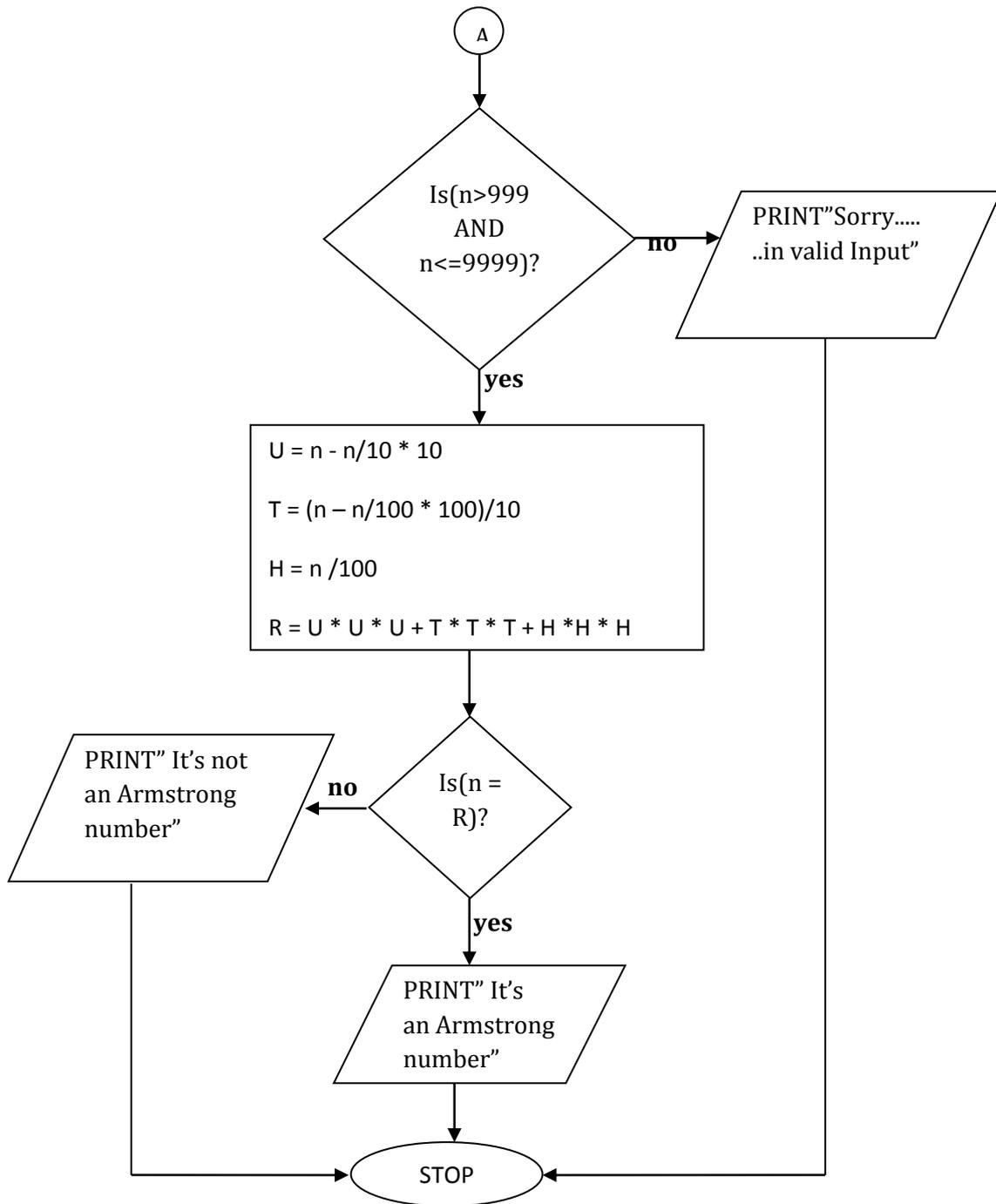
```
void main ()
{
    cout<<"\n hello world";
}
```

2. variable declaration statement: Every variable must be declared just once before it is used. Declaring a variable means specifying the data-type of the corresponding variable. **Example: `char c`;** In the given example, `c` is a variable of type character; capable of storing any single character value which needs one byte of memory space for storage.
3. Decrement operator: The decrement operator is represented as `--` in C++. It decrements the value of the operand by 1 and stores the decremented value in the same operand. Example: `int a =6; --a;` In this example, the value of `a` will be decremented to 5 and this value will be stored in variable `a` itself. As given in the syntax/example, it has just one operand hence; it is termed as unary operator. **But, the operand must always be a variable.**

Q4.B. Draw a flowchart that reads a 3-digit positive integer. Check whether this integer is an Armstrong number or not. [NOTE: AN ARMSTRONG NUMBER IS A NUMBER SUCH THAT THE SUM OF CUBE OF ITS DIGITS IS EQUAL TO THE NUMBER ITSELF. EG. $153 = 1^3 + 5^3 + 3^3 = 153$]

[4]





Q5. Attempt any two of the following.

[10]

1. Write a program in C++ to read the elements of a 3 x 3 determinant and check whether the determinant value is negative. If not, check whether it is zero or perfect square or neither. Print the result with a suitable message in each case.

```
#include<iostream.h>
#include<conio.h>
#include<math.h>
void main( )
{
    clrscr ( );
    int a, b, c, d, e, f, g, h, i, D;
    cout<<"\n enter the elements of a 3 x 3 determinant";
    cin>>a>>b>>c>>d>>e>>f>>g>>h>>i;
    D = a * (c * i - h * f) - b * (d * i - g * f) + c * (d * h - g * e);
    if (D<0)
        cout<<"\n The determinant value is negative";
    else if(D==0)
        cout<<"\n The determinant value is 0";
    else if(int(sqrt(D)) * int(sqrt(D)) == D)
        cout<<"\n The determinant value is a perfect square";
    else
        cout<<"\n The determinant value is neither 0 nor negative nor a perfect square";
}
```

2. Write a menu-driven program in C++ that performs one of the following tasks as per user's choice.

1. To calculate area of circle
2. To calculate area of square
3. To calculate area of rectangle
4. To calculate area of sphere

```
#include<iostream.h>
#include<conio.h>
void main ( )
{
    clrscr ( );
    int s;
    float r,L;
    cout<<" Enter..... 1. To calculate area of circle"
    <<"\n2. To calculate area of square"
    <<"\n3. To calculate area of rectangle"
```

```

<<"\n4. To calculate area of sphere";
cin>>s;
switch(s)
{
    case 1: cout<<"\n Enter the radius of a circle";
            cin>>r;
            cout<<"\n the area of circle is "<<3.14*r*r;
            break;
    case 2: cout<<"\n Enter the side of a square";
            cin>>r;
            cout<<"\n the area of square is "<<r*r;
            break;
    case 3: cout<<"\n Enter the length and breadth of a rectangle";
            cin>>L>>r;
            cout<<"\n the area of rectangle is "<<L*r;
            break;
    case 4: cout<<"\n Enter the radius of a sphere";
            cin>>r;
            cout<<"\n the area of sphere is "<<4*3.14*r*r;
            break;
    default:cout<<"\n Sorry.. Invalid input";
}
}

```

3. Write a C++ that program that reads coefficients of a quadratic equation. It must find and print the root values and nature of the roots.

```

#include<iostream.h>
#include<conio.h>
#include<math.h>
void main ( )
{
    clrscr ( );
    float A, B, C, D, X1, X2;
    cout<<"\n Enter the coefficients of a quadratic equation";
    cin>>A>>B>>C;
    D = B * B - 4 * A * C;
    if (D>0)
    {
        X1 = (- B + sqrt(D))/( 2 * A);
        X2 = (- B - sqrt(D))/( 2 * A);
        cout<<"\n Roots are real and unequal and values are : "<<X1<<" and "<<X2;
    }
}

```

```
    }
    else if(D==0)
    {
        X1 = - B / (2 * A);
        cout<<"\n Roots are real and equal and its value is : "<<X1;
    }
    else
        cout<<"\n Imaginary Roots";
}
```

SHRI T.P. BHATIA COLLEGE OF SCIENCE

I SEMESTER EXAMINATION 2017-2018

MARKS: 50

COMPUTER SCIENCE- I

TIME: 3 HRS

DATE: 06/10/2017

CLASS: XI-K

Q1.A. Select the most appropriate option and rewrite the following statement s. [5]

- Which of these is not a valid expression in C++?
a) $a / 3$ **b) $a \% 3.0$** c) $a / 3.2$ d) All of these
- Which of these is a valid constant value in C++?
a) 12,300 **b) 'H'** c) 'AB' d) All of these
- The execution of a C++ program starts from
a) The very first line of the program **b) main() function**
c) The first statement written within the main () function d) None of these
- A keyword is
a) A special word with a specific meaning. b) Always specified in small case characters
c) Already defined to the compiler **d) All of these**
- Which of these is a valid variable declaration statement in C++?
a) `int a, int b;` **b) `float x; float y;`** c) `float c, int m;` d) `int a , b`

Q1.B. State whether the following statements are True or False. [5]

- [], { } and () cannot be used in arithmetic instructions in C++.
ANS: FALSE
- '-' operator is a unary as well as binary operator in C++.
ANS: TRUE
- Operators having equal precedence are always evaluated from left to right in C++.
ANS: FALSE
- A condition must be always specified with 'else' in if-else structure in C++.
ANS: FALSE
- C++ programs are converted into binary language with the help of software- 'Compiler'.
ANS: TRUE

Q2.A. Answer any two of the following. [6]

- Explain insertion and extraction operators, in C++, with an example.

ANS: a) The operators used for input and output are >> and << called as extraction operator and insertion operator.

b) In C++, cin represents the console input, >> operator extracts the value from console input and stores the value in the variable x to its right. **Example:** **cin>>a;**

c) The << operator takes the statement/value to its right and inserts on the console output represented by cout. **Example:** **cout<<"My roll no. is "<<roll;**

2. What do you mean by a variable in C++? State four distinct rules of naming a variable in C++.

ANS: Variable: A variable is a name given to memory location(s) that store value(s).

Rules of naming a variable

1. A variable name is a combination of alphabets, digits or/and underscore.
2. The maximum length of the name should not be more than 32 characters.
3. The first character in the name must not be a digit.
4. Existing keywords must not be used as variable names.

3. Write an algorithm to read coordinates of center of a circle and its radius as well. Also, read coordinates of a point from the user. Check whether this point lies within the circle, outside the circle or on the circle.

ANS: 1. PRINT "Enter coordinates of a center of a circle and radius as well"

2. READ x, y, r

3. PRINT "Enter coordinates of a point as well"

4. READ x1, y1

5. $d = \sqrt{(x1 - x) * (x1 - x) + (y1 - y) * (y1 - y)}$

6. if (r = d) then

 PRINT "The given point lies on the circle"

 Else if (r>d) then

 PRINT "The given point lies inside the circle"

 Else

 PRINT "The given point lies outside the circle"

7. END

Q2.B. Write an Algorithm to read coefficients of a quadratic equation from the user. Find and print the roots and nature of the roots of the given equation. **[4]**

ANS: 1. PRINT "Enter the coefficients of a quadratic equation"

2. READ a, b, c

3. $d = b * b - 4 * a * c$

4. If(d = 0) then

$$X = -b / (2 * a)$$

PRINT "The roots are real and equal", X

Else if (d>0) then

$$X1 = (-b + \sqrt{d}) / (2 * a)$$

$$X2 = (-b - \sqrt{d}) / (2 * a)$$

PRINT "The roots are real and unequal X1= ", X1, " and X2 = ", X2

Else

PRINT "The roots are imaginary"

5. END

Q3. Attempt any five of the following.

[10]

1. Write two distinct C++ statements for the expression $C = X^4$.

ANS: 1) $C = \text{pow}(X, 4)$

2) $C = X * X * X * X$

2. Write an algorithm to read temperature in Celsius from the user. Convert and print the given temperature in Fahrenheit. [Note the formula. $C = (F - 32)X \frac{5}{9}$]

ANS: 1. PRINT "Enter the temperature in Celsius"

2. READ C

3. $F = C * 9 / 5 + 32$

4. PRINT "The given temperature in Fahrenheit is ", F

5. END

3. Analyze the following set of C++ codes and explain the difference in the output generated with the given codes.

a) `cout<<"\n Hi";`

`clrscr();`

b) `clrscr();`

`cout<<"\nHi";`

ANS: a) the code print the statement "Hi" on the output screen and immediately the clear screen function is called because of which the output generated on output screen is cleared off.

b)the code first clears the output screen which had the output of previous execution and immediately displays the statement “Hi” represented by the following cout statement.

4. State any two (distinct) differences between = and == operators in C++.

= (Assignment operator)	==(Relational operator)
This operator assigns (stores) the value of right hand side operand/expression to the left hand side operand.	This operator compares the left operand with right operand for equality and returns true if they are equal
The left operand must always be a variable	The left and right operand both can be in any form (variable/constant/expression)
Eg. <code>A = 8 * r</code>	Eg. <code>8 == A</code>

5. Explain compile time and run time variable initialization in C++, with an example.

ANS: In compile initialization the variable’s initial value is preset in the program itself and known to the compiler at the time of compilation. **Example: `sum = 0;`**

In run time initialization the variable’s initial value is not predetermined. It can be assigned by the user at the time of execution of program or it could be set by an expression which is not known to the compiler beforehand. **Example: `area = 3.14 * r * r;`**

6. State and explain any two backslash sequence character in C++, with an example.

ANS: ‘\n’, ‘\t’ and ‘\a’ are the backslash sequences in C++.

\n is a new line character sequence. It is used to break the current line and go to the first column in the next line.

\t is a horizontal tab character. It is used to give multiple spaces (generally 8 character spaces) from the current position.

\a is a sound character. It is used to generate an alert bell.

Example: `cout<<“\n Name:\tSachin Tendulkar”;`

Q4.A. Attempt any three of the following.

[6]

1. Write and explain the syntax of a conditional operator in C++.

ANS: the syntax of conditional operator:

Expression 1 ? Expression 2 : Expression 3

If the condition is Expression 1 evaluates to true value then Expression 2 is executed else Expression 3 will be executed.

2. What is the use of 'default' and 'break' keywords in switch block, in C++?

ANS: default: if the all the mentioned cases' value doesn't match with switch value then default case statements will be executed. It is like the else block in if-else structure.

break: it is used to stop the further execution of switch-case block and transfer the control of execution to the statement following the switch block.

3. Explain cascading in C++, with an example.

ANS: Combining multiple input/output (cin/cout) statements into single input/output (cin/cout) statement is called as *cascading*.

Example: cout<<"My roll no. is";
 cout<<roll;

The above 2 cout statements can be combined as

cout<<"My roll no. is "<<roll;

4. What is the difference between 'signed int' and 'unsigned int' in C++, with an example?

ANS: a variable of type signed int can store a -ve , 0 or a +ve integer value within the range of -32768 to + 32767. A variable of type unsigned int can store a 0 or +ve integer value within the range of 0 to 65536.

Q4.B. If a = 10, b = 12, c = 0, find the values of the expressions in the following table:

[4]

Expression	Value
a != 6 && b > 5	1
a == 9 b < 3	0
!(a < 10)	1
!a > 5 && c	0

Q5. Answer any two of the following.

[10]

1. Write a C++ program to read marks obtained in 6 subjects by a student from the user. Calculate and print total marks, percentage and Class (as per the given criteria) obtained by the student with a suitable message. [NOTE: Distinction with first class for 75% or above, First class for 60% or above but less than 75%, Second class for 45% or above but less than 60%, Pass class for 35% or above but less than 45%; otherwise failed for percentage below 35 or students scoring less than 35 in any one of the subjects.]

```
void main()
{
    clrscr();
    int p, c, m, e, cs1, cs2, total;
    float per;
    cout<<"\n enter the marks obtained in 6 subjects";
    cin>>p>>c>>m>>e>>cs1>>cs2;
    total = p + c + m + cs1 + cs2 + e;
    per = total / 6.0
    if (per<35 || p<35 || c<35 || m<35 || e<35 || cs1<35 || cs2<35)
        cout<<"\n total marks ="<<marks<<" percentage = "<<per<<"\n the student has
        failed";

    else if (per>=75)

        cout<<"\n total marks ="<<marks<<" percentage = "<<per
        <<"\n the student has passed with distinction";

    else if (per>=60)

        cout<<"\n total marks ="<<marks<<" percentage = "<<per
        <<"\n the student has passed with first class";

    else if (per>=45)

        cout<<"\n total marks ="<<marks<<" percentage = "<<per
        <<"\n the student has passed with second class";

    else

        cout<<"\n total marks ="<<marks<<" percentage = "<<per
        <<"\n the student has passed with pass class";
```

```
}
```

2. Write a program in C++ to read a character from the user. Check and print whether the given character is an alphabet, digit or a special symbol.

```
void main()
```

```
{
```

```
clrscr();
```

```
char n;
```

```
cout<<"\n enter a character value";
```

```
cin>>n;
```

```
if ((n>='A' && n<='Z') || (n>='a' && n<='z'))
```

```
    cout<<"\n the given character is an alphabet";
```

```
else if (n>='0' && n<='9')
```

```
    cout<<"\n the given character is a digit";
```

```
else
```

```
    cout<<"\n the given character is a special symbol";
```

```
}
```

3. Write a C++ program to read four distinct integer values from the user. Find and print the greatest integer from the given ones. It must give an appropriate message if the values are not distinct.

```
void main()
```

```
{
```

```
clrscr();
```

```
int a, b, c, d;
```

```
cout<<"\n enter four distinct integer values";
```

```
cin>>a>>b>>c>>d;
```

```
if (a ==b || a==c || a==d || b==c || b ==d || c==d)
```

```
    cout<<"\n the given integers must be distinct";
```

```
else if (a>b && a>c && a>d)
```

```
    cout<<"\n the greatest integer is "<<a;
```

```
else if (b>c && b>a && b>d)
```

```
    cout<<"\n the greatest integer is "<<b;
```

```
else if (c>a && c>b && c>d)
```

```
    cout<<"\n the greatest integer is "<<c;
```

```
else
```

```
    cout<<"\n the greatest integer is "<<d;  
}
```

SHRI T.P.BHATIA COLLEGE OF SCIENCE

I SEMESTER EXAMINATION 2015-16

MARKS: 50

COMPUTER SCIENCE- I

CLASS: XI C

DURATION: 3HRS

DATE: 30/10/2015

Q.1.A. Select the most appropriate option and rewrite the statement.

[5M]

- The preprocessor directives need to be written _____ the main function
 - Before**
 - Inside
 - after
 - both b and c
- Which of the following is the correct order of evaluation for the below expression?

$$z = x + y * z / 4 \% 2 - 1$$
 - */%+-=**
 - /*% - +=
 - = */% + -
 - * % / - +=
- How many words can be read by a single cin statement in C++ ?
 - Multiple
 - One**
 - Depends on what user has entered
 - Depends on compiler
- Which of the following is required to write and run a C++ program
 - Compiler
 - Text editor
 - Operating system
 - All of the above**
- Which of the following is an incorrect statement in C++ ?
 - x=6
 - 6=x;
 - Both a and b**
 - None of these

Q.1.B. Find and rectify the errors in the following C++ code

[4M]

```
1. void main
{
    float a=12.23;b=12.52;
    IF(a=b)
    {    cout<<" a and b are equal";
    }
}
```

Sol:

```
void main()
{
    float a=12.23,b=12.52;
    if(a=b)
    {    cout<<" a and b are equal";
    }
}
```

```
2. void main()
{
    Int a,b,c=20;
    a=b=c;
    if(a!20)
        cout<<" Hello"
    elseif(b==20)
        cout<<" Hii";
}
```

Sol:

```
void main()
{
    int a,b,c=20;
    a=b=c;
    if(a!=20)
        cout<<" Hello";
    else if(b==20)
        cout<<" Hii";
}
```

```
}
```

Q.2. Attempt ANY 5 of the following.

[10M]

1. What is Type Casting ? Explain how a 'float' value be type casted into an 'int' value

Sol: Type Casting is a process in which the datatype of a particular variable is changed to some other datatype temporarily i.e just for a single operation .Eg : An 'int' datatype variable can be converted to 'float'

There are two different syntax for type conversion

```
float x =10.3;
cout<<int(x);
OR
float x = 10.3;
int a=x;
cout<<x;
```

2. Give two differences between unary and binary operators. [Note: Include an example]

Sol:

Unary operators	Binary operators
1) These operators need only one operand to perform an operation	1) These operators need two operands to perform an operation.
2) Associativity of unary operator is from Right to Left.	2) Associativity of binary operator can be from Right to Left or Left to Right.
3) Eg: ++ (increment operator), - (unary minus operator) etc..	3) Eg: + (Addition), ==(Equal to) etc..

3. Write a C++ statement that

a) Prints " You are an adult" if age is from 18 to 60

Sol:

```
if(age>=18 && age<=60)
    cout<<"You are an adult";
```

b) Prints whether the value of a variable 'c' is a vowel.

Sol:

```
if(c=='a' || c=='e' || c=='i' || c=='o' || c=='u' || c=='A' || c=='E' ||
c=='O' || c=='U')
    cout<<"It is a vowel";
```

4. Describe the two ways to include comments in a C++ program

Sol:

The two ways to include comments in C++ program are:

1) Single line comments: Any sentence after the double forward slash(//) is called single line comment. It is only for that particular line.

Eg: //Welcome to C++ programming

2) Multiple line comments: To give multiple line comments, the comment should start with a delimiter /* and end with the delimiter */

Eg: /* Welcome to C++ programming
This is my first program */

5. What will be the output of the following program

```
void main()
{
    int a=5, b;
    b=++a+a+++a;
    cout<<a<<<"\t";
    cout<<b;
}
```

Sol:

The output of the above program will be:

7 18

6. Construct a logical expression to represent each of the following conditions :

a) n is divisible by 3 but not by 30

Sol: $n \% 3 == 0 \ \&\& \ n \% 30 != 0$

b) answer is either 'n' or 'N'

Sol: $ans == 'n' \ || \ ans == 'N'$

Q.3. Attempt ANY 4 of the following.

[12M]

1. Define the term Variable. Are the following C++ statements same or different? Explain.

```
float a;
float a=1.5;
```

Sol:

A variable is an identifier that is used to represent data item /items i.e a numerical quantity, character or a string value.

The given C++ statements are different

```
float a;
```

This statement is just a variable declaration statement in which the variable 'a' will hold a garbage (unknown) value while the next C++ statement i.e

```
float a=1.5;
```

is variable declaration as well as variable initialization in which the variable 'a' will hold a float value

1.5.

2. Explain insertion and extraction operators in C++, with an example of each.

Sol:

Insertion Operator: The Insertion operator '<<' inserts the data that follows it into the stream (cout i.e. console output screen) that precedes it.

Eg : `cout<<" Welcome to C++ programming";`

Extraction Operator: The Extraction operator '>>' is used to extract the data given as an input. The input given will be stored in a variable that follows it (written after the extraction operator)

Eg : `cin>>a;`

3. Write an algorithm to read values of three sides of a shape. Check whether these sides form a valid triangle, if so also check whether it forms an equilateral, isosceles or scalene triangle.

Sol:

1. PRINT " Enter three sides of a shape"

2. READ d1,d2,d3

3. IF(d1+d2>d3 AND d2+d3>d1 AND d1+d3>d2)

IF(d1==d2 AND d2==d3)

PRINT " The given sides form an equilateral triangle"

```

ELSE IF(d1==d2 OR d2==d3 OR d1==d3)
    PRINT” The given sides form an isosceles triangle”
ELSE
    PRINT “ The given sides form a scalene triangle”
ELSE
    PRINT ”The given sides do not form a valid triangle”
4. END

```

4. Enlist 3 basic datatypes used in C++ with range of values and bytes they occupy in memory.

Sol:

Datatype	Range	Bytes
char	-128 to 127	1
int	-32768 to 32767	2
float	-3.4e38 to +3.4e38	4
double	-1.7e308 to +1.7e308	8

5. Write and explain the syntax of switch statement. Also state 2 disadvantages of switch- case .

Sol:

The syntax of switch- case statement is as follows:

```
switch(expression or variable)
```

```

{
    case value_1:
        statements;
        break; //optional
    case value_2:
        statements;
        break;
    case value_n:
        statements;
        break;
    default:
        statements;
}

```

Explanation:

The switch statement is a selection statement. Followed by the keyword 'switch', a variable or an expression is placed in the parenthesis . The body of the switch statement is enclosed in braces .It consists of 'Case Labels', which take the keyword 'case' followed by a constant value. The case labels are followed by a colon. When a switch variable or expression is evaluated , the control is transferred to the case label whose constant is equal to the derived value and then all the statements followed by the case label are executed. The break statement is used to terminate the execution of switch block. Special case label 'default' is used if the expression does not match any of the case label

Disadvantages:

- 1) A float expression cannot be tested using switch.
- 2) Multiple cases cannot use same expressions.
- 3) Cases can never have variable expressions.(Eg: case x+4: is invalid)

Q.4A Define the following terms with respect to C++ (ANY 5)

[5M]

a. **Keywords :**

Keywords are the special words, which are designated to do a specific task and whose meaning is already known to the compiler.

b. Compiler :

Compiler is a computer program that accepts a program written in high level language as input and generates a corresponding machine language program as output. It translates the entire program into machine language before executing any of the instruction.

c. Identifiers :

Identifiers are names that are given to various program elements, such as variables, functions etc.. Identifiers consists of letters, digits and underscore, except that the first character must not be a digit.

d. Cascading :

Combining multiple cout/cin statements into a single cout /cin, in c++, is called as Cascading

e. Literals:

Literals refer to fixed values that the program cannot alter. Literals can be of any of the basic data types and can be divided into Integer Numerals, Floating-Point Numerals, Characters and string.

f. Escape sequence :

Certain non-printing characters can be expressed in terms of escape sequences. An escape sequence always begins with a backward slash and is followed by one or more characters eg. '\n' as a newline character

Q.4B. Attempt ANY 1 of the following

[4M]

1. a) Justify the following : Every C++ program should have a main function.

Sol :

1) When a C++ program is under execution, the control of the computer is passed over to that program. The key point is that the computer needs to know where inside your program the control needs to be passed. In the case of a C++ program, it's the main() function that the computer is looking for.

2) At a minimum, the main() function looks like this:

```
main() { }
```

Like all C++ language functions, first comes the function's name, main, then comes a set of parentheses, and finally comes a set of braces, also called *curly braces*.

If a C++ program contains only this line of code, it can be executed. It won't do anything, but that's perfect because the program doesn't tell the computer to do anything. Even so, the computer found the main() function and was able to pass control to that function — which did nothing but immediately return control right back to the operating system.

[Note: Only 1st point is required for as the solution to the justification. The 2nd point is just a supporting explanation to it.]

b) Give 2 points of difference between '!' and '!=' operators.

Sol:

- 1) '!' is a Logical negation operator while '!=' is a Relational Operator
- 2) '!' is an unary operator while '!=' is a binary operator
- 3) '!' operator causes an expression that is originally true to become false or vice versa while '!=' checks whether its first operand is not equal to the second operand.

2. Write four different expressions which increments the value of integer variable x

by 1.

Sol:

The 4 different expressions are :

- 1) $x=x+1$;

- 2) x++;
- 3) ++x;
- 4) x+=1;

Q.5. Attempt ANY 2 of the following.

[10M]

1. Write a Menu-driven program in C++, to read a measurement from the user in Feet (F) or in Inches (I), as per the user's choice. The program converts the given measurement from one unit into the other. Display the converted value along with the given value, with a suitable message.

Sample Output:

```
Convert 1. Feet to Inches           2. Inches to Feet
Enter your choice      1
Enter the measurement in Feet      8
The given measurement 8 feet is equivalent to 96 inches.
```

Solution:

```
#include<iostream.h>
#include<conio.h>
void main()
{
    clrscr();
    int f,i,ch;
    cout<<"\n Convert : \t 1.Feet to Inches \t 2.Inches to Feet";
    cout<<"\n Enter your choice : ";
    cin>>ch;
    switch(ch)
    {
        case 1:
            cout<<"\n Enter the measurement in Feet : ";
            cin>>f;
            cout<<"\n The given measurement "<<f<<" Feet is           equivalent to "
            <<f*12<<" inches";
            break;
        case 2:
            cout<<"\n Enter the measurement in Inches : ";
            cin>>i;
            cout<<"\n The given measurement "<<i<<" inches is           equivalent          to
            "<<(i/12)<<" feet";
            break;
        default:
            cout<<" Enter a valid option";
    }
}
```

2. Write a program in C++ to read co-ordinates of 3 points from the user. Check if these points fall on a straight line or not.

Solution:

```
#include<iostream.h>
#include<conio.h>
void main()
{
    clrscr();
    int x1,y1,x2,y2,x3,y3,s1,s2, s3;
    cout<<"\n Enter the co-ordinates of first point";
    cin>>x1>>y1;
    cout<<"\n Enter the co-ordinates of second point";
```

```

cin>>x2>>y2;
cout<<"\n Enter the co-ordinates of third point";
cin>>x3>>y3;
s1 = sqrt((x2-x1) * (x2-x1) + (y2-y1) * (y2-y1));
s2 = sqrt((x3-x2) * (x3-x2) + (y3-y2) * (y3-y2));
s3 = sqrt((x3-x1) * (x3-x1) + (y3-y1) * (y3-y1));
if(s1 + s2 ==s3 || s1 + s3 == s2 || s2 + s3 == s1)
    cout<<"\n The given 3 points fall on the straight line";
else
    cout<<"\n The given 3 points don't fall on straight line";
}

```

3. Write a program in C++ to read a 4-digit integer from the user. Check whether the entered integer is a 4-digit or not. If so, then check whether the integer is palindrome number or not. Print the result with a suitable message.

Solution:

```

#include<iostream.h>
#include<conio.h>
void main()
{
    clrscr();
    int u,t,h,th,rev,x;
    cout<<"\n Enter a four digit number";
    cin>>x;
    if (x>=1000 && x<=9999)
    {
        u=x% 10;
        t=(x/10)% 10;
        h=(x/100)% 10;
        th=x/1000;
        rev=th+h*10+t*100+u*1000;
        if (rev==x)
            cout<<"\n The entered number is a palindrome          number";
        else
            cout<<"\n The entered number is not a palindrome      number";
    }
    else
        cout<<"\n Enter a valid four digit number";
}

```