

Sr. No	Name of Resource	Specification	Quantity	Remarks
1.	Hardware : Computer System	Computer (i3-5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2.	Operating system	Windows/LINUX		
3.	Software	Turbo C++ Version 3.0 or any other		

IX**Precautions**

- Handle computer system and peripherals with care.
- Follow safety practices.

X

S. No	Name of Resource	Specification
1	Computer System with broad specifications	hp 280 g2 , i3 processor
2	Software	Turbo C++
3	Any other resource used	Google Chrome

XI**Result (Output of the Program)**

.....Successfully executed the programs using.....
.....operator overloading & expression.....

XII**Practical Related Questions**

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- Constant variables can be created in C++ by using _____.
 - const
 - #define
 - Both a& b
 - None of these
- State output of the following code:

```
#include <iostream.h>
void main()
{
    typedef int num;
    num a = 10, b = 15;
```

XIII**Exercise**

Attempt Q1 or Q2 and Q3 a or b from the following:
(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- Write a C++ program to evaluate the following expressions
$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$
- Write a C++ program to demonstrate the use of operator precedence.

- Complete the given table:

Program Code	Write & Justify Output
a) #include<iostream.h> #define PI 3.14159 int main () { float r=2; // radius float circle=(r*PI); // area of circle cout<<circle; } return 0;	OP = 12.56 Value to be justified Hence multiplication of circle=(r*PI) will be 12.56 as r=2, PI=3.14

float r=2; // radius
float circle=(r*PI); // area of circle
cout<<circle;
return 0;

As r=2, PI=3.14
will be 12.56
This is assigned to
variable circle.

num c = a + b + a - b;
Output = 20

- Regarding following statement which of the statements is true? const int a = 100;
 - Declares a variable a with 100 as its initial value
 - Declares a construction a with 100 as its initial value
 - Declares a constant a whose value will be 100
 - Constructs an integer type variable with a as identifier and 100 as value

- Which of the following statement is not true about preprocessor directives?
 - These are lines read and processed by the preprocessor
 - They do not produce any code by themselves
 - These must be written on their own line
 - They end with a semicolon

```
a) #include<iostream.h>
#include<conio.h>
void main()
{
    clrscr();
    float res;
    float f1=15.5, f2=2.5;
    float f3=f1*f2;
    res = (int)f3;
    cout<<res<<endl;
    res = (int)(f1*f2);
    cout<<res<<endl;
    cout<<res<<endl;
    cout<<res<<endl;
    getch();
}
```

The o/p will be -
 7 7 7.5
 because when eqn
 becomes (int)f1 / (int)f2;
 it converts
 floating value into int
 same will happen with
 float (F1/F2);
 result (F1/F2);
 but finally there is no float
 type conversion hence o/p
 is an floating value.

(Space for Answers)

Name: Gaikwad Supriya Yuvraj

Class: S. Y CSE

Roll No: 10

CODE

```
#include<stdio.h>
```

```
#include<iostream.h>
```

```
Q.1
# include <conio.h>

Void main()
{
    clrscr();
    cout<<"Enter values for ab, &c ";
    cin >> a >> b >> c;
    const r = b * b - 4 * a * c;
    r1=(a+b)*2;
    r2=c+a*b;
    r3=a*c*a+b;
    cout<<"value of r1 = " << r1;
    cout<<"value of r2 = " << r2;
    cout<<"result is=" <<r3<<endl;
    cout<<"result is=" <<r2<<endl;
    getch();
}
```

OUTPUT

Q1

Enter integer no's:9743

result is=448

result is=67

result is=314

① ~~Malvi~~

www.msbtesolution.xyz

Malvi

Malvi

Malvi

Malvi

Malvi

Malvi

Malvi

If-else Statement :- In general it can be used to execute one block of statement among two blocks, if and else are the keyword in C++.

Syntax

```
if(condition)
{
    statements
}
else
{
    statements
}
.....
```

Switch Statement:-
A switch statement work with byte, short, char and int primitive data type, it also works with enumerated types and string.

Syntax

```
switch(expression/variable)
{
    case value:
        //statements
    // any number of case statements
    break; //optional
    default: //optional
    //statements
}
```

VIII Resources required

Sr. No.	Name of Resource	Specification
1	Computer System with bread specifications	window 7 intel i3 3220 CPU, 3.30 GHz/6gbt
2	Software	Turbo C++
3	Any other resource used	internet software

- IX Precautions**
- Handle computer system and peripherals with care.
 - Follow safety practices.

X Resources used

S. No.	Name of Resource	Specification
1	Computer System with bread specifications	window 7 intel i3 3220 CPU, 3.30 GHz/6gbt
2	Software	Turbo C++
3	Any other resource used	internet software

XI Result (Output of the Program)

Be successfully executed program to implement decision making statements

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Observe the following block of code and determine what happens when x=2?

```
switch (x) {
    case 1:
        cout<<"x is 1";
    case 2:
        cout<<"x is 3, so jumping to third branch"; goto thirdBranch;
    default:
        cout<<"x is not within the range, so need to say
        Thank You!";
}
```

2. Conditional operator in C++ is a

- a) Unary operator b) Binary operator c) Ternary operator d) None of them

3. Which of the following is selection statement in C++?

- a) break b) goto c) exit

- d) switch

[Space for Answers]

- Q.1 The 'break' control statement is not used in the program code. So that means even one condition satisfied with given parameter will not break the whole switch

Exercise

1. **Void main()**

```
int year;
cout << "Enter any year" << endl;
```

```
cin >> year;
if (year % 4 == 0)
    cout << "This year is leap year" << endl;
```

```
else
    cout << "This year is not leap year" << endl;
```

```
cout << "This year is not leap year" << endl;
```

```
getch();
```

Q.1 Loop case after satisfied case will also execute due to non-use of break statement.

So after case 3, control will go to the 'thirdbranch' statement.

Q.2 Condition operator in if is also called as ternary operator.

Q.3 Switch is the selection statement other than control statement.

XIII Exercise:
Attempt Q1 or Q2 or Q3 and Q.4 a,b or a,c from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program check leap year.

2. Write a C++ program armstrong numbers between two integers.

3. Write a C++ program to check whether a number is palindrome or not.

4. Complete the given table:

Program Code	Output	Write & justify
a) #include <iostream.h> void main() { int a = 10; if (a < 15) { time; cout << a; goto time; } break; }	Output 3! = 6	break statement is at wrong position. It should not there in the program. If we would have to uses break then we should give some condition to stop the looping. We should change place at goto statement.

Program Code	Output	Write & justify
b) #include <iostream.h> long factorial (long a) { if (a > 1) return (a * factorial (a - 1)); else return (1); }	Output 3! = 6	Output 3! = 6

Program Code	Output	Write & justify
c) #include <iostream.h> void main() { int percentage; cout << "Enter the percentage : "; cin >> percentage; switch (percentage / 10) { case 10: case 9: cout << "You have got grade A+" << endl; break; case 8: cout << "You have got grade A" << endl; break; case 7: } }	Output 3! = 6	In program code there is no function to hold SLP screen. If we write screen function then the program will be correct. It checks if we enter percentage greater than 90 then

```

cout << "You have got grade E" << endl;
break;
case 6:
cout << "You have got grade E" << endl;
break;
case 5:
cout << "You have got grade C" << endl;
break;
default:
cout << "You have got grade D" << endl;
break;
}

```

Two switch cases will execute due to case 6 will be carried further calculation will occur as switch cases.

(Space for Answers)

```

3.
void main()
{
    int num1, num2, i, num, digit, sum;
    cout << "Enter first number: ";
    cin >> num1;
    cout << "Enter second number: ";
    cin >> num2;
    cout << "Armstrong numbers between " << num1 << " and " <<
    num2 << " are: " << endl;
    for(i = num1; i <= num2; i++)
    {
        num = i;
        sum = 0;
        for(; num > 0; num /= 10)
        {
            digit = num % 10;
            sum = sum + digit * digit * digit;
        }
        cout << "No. is not Palindrome " << endl;
    }
    cout << " No. is Palindrome " << endl;
}

```

Name: Gaikwad Supriya Yuvraj.
Roll no.: 10
Class: SY CSE.

SOURCE CODE

```

cout << "You have got grade E" << endl;
break;
case 6:
cout << "You have got grade D" << endl;
break;
case 5:
cout << "You have got grade C" << endl;
break;
default:
cout << "You have got grade D" << endl;
break;
}
}

```

~~How switch case will execute output
for case 10⁴ cases will be
done further calculation will
occur by switch cases.~~

(Space for Answers)

Name: Gaikwad Supriya Yuvraj.
 Roll no.: 10
 Class: SY CSE.

SOURCE CODE

```

void main()
{
    int num1, num2, i, num, digit, sum;
    cout << "Enter first number: ";
    cin >> num1;
    cout << "Enter second number: ";
    cin >> num2;
    cout << "Armstrong numbers between " << num1 << " and " <<
    num2 << " are: " << endl;
    for(i = num1; i <= num2; i++)
    {
        sum = 0;
        num = i;
        for(; num > 0; num /= 10)
        {
            digit = num % 10;
            sum = sum + digit * digit * digit;
        }
        cout << " No. is not Palindrome " << endl;
    }
}

```

```
if(sum == i){  
    t  
    cout << i << endl;
```

Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		
1	Logic formation	70%
2	Appropriate use of Decision making statements (if-else , switch)	20%
3	Debugging ability	20%
4	Follow ethical practices.	10%
Product related (15 Marks)		
5	Expected Output	30%
6	Timely Submission of report	10%
7	Answer to sample questions	10%
Total (50 Marks)		100%

List of Student Team Members

1. Gaikwad komal
 2. Gaiikwad Supriya
 3. Ghunke Mukti
 4. Gurav Prathmesh

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related(15)	Total(50)	
93	13	66	<i>[Signature]</i>

VIII Resources required

S. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating System	Windows / LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- Handle computer system and peripherals with care.
- Follow safety practices.

X Resources used

S. No.	Name of Resource	Specification
1	Computer System with broad specifications	i5 Processor
2	Software	Turbo C++
3	Any other resource used	private Internet

XI Result (Output of the Program)

We successfully executed programs to implement Control Structure.

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- Observe the following block of code and determine what happens when $x=2$?

```
#include<iostream.h>
void main()
{
    int a,d,n,sum,term=0;
    cout<<"Enter the first term, common difference,"
```

2. Which loops needs a semi colon after?
 a) for b) do c) while
 3. In the while statement, while($x < 100$).... when does the statement controlled by the condition execute?
 ✓ When x is less than one hundred
 b. When x is greater than one hundred
 c. When x is equal to one hundred
 d. While it wishes

(Space for Answers)

Q1: There is no any variable x is used throughout the program hence it does not change so it can't harm the program output if we change it.

The o/p of current program -

{ input - first term a = 1

common difference d = 2

number of terms n = 3

o/p = term 1 3 5

The sum of A.P is 5 . 3

but if we change the statement

$c = sum = term'$ to 'sum = sum + term' then final o/p statement will be :

The sum of A.P is 9.

```
<<"and the number of terms to be summed"
<<"respectively.\n";
Cin>>a>>d>>n;
cout<<"\nThe terms are ";
do
{
    term= a + (i-1)*d;
    sum+=term;
    cout<<term<<" ";
} while (i<=n);
cout<<"\nThe sum of A.P. is "<<sum;
}
```

Name: Gaikwad Supriya Yuvraj

Roll no.: 10

Class: SYCSE.

-----Source Code-----

```
#include<iostream.h>
```

```
void main()
```

```
int i;
```

```
for(i = 100; i >= 1; i++)
```

```
{
```

```
printf("%d \t",i);
```

```
}
```



XIII Exercise

Attempt any 2 from 1-4 and Q5 a or b from the following:

(Note: Use Point will 'o' X and Aifl to 'X' if you programming extension use blank space after and attach file names if required.)

1. Write a C++ program to print multiplication table of 7.
2. Write a C++ program to print 10! to 1.
3. Write a C++ program which will print half pyramid pattern of natural numbers.

Output
100 99 98 97 96 95 94 93 92 91 90 89 88 87 86 85 84 83 82 81 80 79
78 77 76 75 74 73 72 71 70 69 68 67 66 65 64 63 62 61 60 59 58 57
56 55 54 53 52 51 50 49 48 47 46 45 43 42 41 40 39 38 37 36 35 34
33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12
11 10 9 8 7 6 5 4 3 2 1
7 8 9 10
11 12 13 14 15

4. Write a C++ program, ask the user to enter the number of rows to print the

Pyramid of stars (*)

* * * * *
* * * * *
* * * * *

© Galic

Name: Gaikwad Supriya Yuvraj

Roll no.: 10

Class: SYCSE

Q.2 'do' loop needs semicolon at the end of the line
'while' keyword.

```
#include<stdio.h>
```

```
void main()
```

int'l

卷之三

1

```
printf("%d \\t", i);
```

三

III Exercise

Attempt any 2 from Q4 and Q5 or D from the page

Note: Use Panel VIII & X and XIII to XV for all relevant

1 Write a C++ program to print 1 to 100 in descending order. 100 99 98 97 96 95 94 93 92 91 90 89 88 87 86 85 84 83 82 81 80 79
2 Write a C++ program which will print half pyramid pattern of natural numbers. 78 77 76 75 74 73 72 71 70 69 68 67 66 65 64 63 62 61 60 59 58 57

卷之三

卷之三

1954-1960
THE FEDERAL BUDGET

78910

1123 HAN

Pyramid of stars (9) 10003 19

卷之三

卷之三

卷之三

卷之三

Output

6

```
1 void main()
2 {
3     int i, s, r, k=0;
4     clrscr();
5     printf (" enter number of rows");
6     scanf ("%d", & r);
7     for (i=1; i<=r; i++, k=0)
8     {
9         for (s=1; s<=r-i; s++)
10        {
11            printf ("*");
12        }
13        while (k!=2*i-1)
14        {
15            printf ("*");
16            k++;
17        }
18        printf ("\n");
19    }
20 }
```

Output :

enter number of rows 5

```
*****
 * * *
 * * * *
 * * * * *
 * * * * *
```

Q.1 #include <iostream.h>

#include <stdio.h>

#include <conio.h>

void main()

{

int i, no, table = 1;

char c;

cout << "Enter a table number";

cin >> no;

cout << "Table of " << no;

for (i=1; i<10; i++)

{

table = no * i;

cout << i << table;

}

cout << endl;

}

getch();

Output - Enter the table number 7

Table of 7

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

www.msotesolution.xyz

5. Complete the given table:

Program Code	Write & justify Output
<pre>a) #include <iostream.h> int main() { int a, d, n; std::cout << "Enter a, d, n of AP: "; std::cin >> a >> d >> n; std::cout << std::endl << "Terms of AP : a = " << a << std::endl; do { cout << " ", d = " << d << ", n = " << n << std::endl; a = a + d; } while (n-- > 1); std::cout << std::endl; }</pre>	<p>We can't use Scope resolution operator because there is no any global declaration & no need to use Scope resolution operator we can write program without :: operator</p>
<pre>b) #include <iostream.h> #include <stack.h> int main() { std::stack<char> s; // Pushing elements into stack for (char c = 'a'; c <= 'f'; c++) s.push(c); while (!s.empty()) { std::cout << "Top element of stack" << << s.top(); s.pop(); std::cout << "\n" << std::endl; } std::cout << "Stack is empty!" << std::endl; }</pre>	<p>Justification It is unable to open Stack.h header file</p>

(Space for Answers)

Exercise

3. void main()

```
{}
int i, j, k=1;
for (i=1; j<5; i++)
    for (j=1; k<1; j++)
        k = k + 1;
```

Practical No. 4: Program to Implement One Dimension Array

```

cout << f << n "f
t t /
} cout << endl;
}
  
```

Suggestions for further Reading

XIV References / Suggestions for further Reading for r-loop

1. <https://www.programiz.com/cpp-programming/iteration-statements>
2. <http://cccomputermoles.com/cpp/control-structures/iteration-statements/do-while-loop/>
3. <https://www.sanfoundry.com/cpp-program-demonstrate-do-while-loop/>
4. <https://codecracker.com/cpp/print-star-pyramid-patterns.htm>

XV Assessment Scheme

Performance indicators	Weightage
Process related(35 Marks)	70%
1 Logic formation	20%
2 Appropriate use of control structure (for, while, do-while)	20%
3 Debugging ability	10%
4 Follow ethical practices.	30%
Product related (15 Marks)	20%
5 Expected Output	10%
6 Timely Submission of report	10%
7 Answer to sample questions	100%
Total (50 Marks)	

List of Students Team Members

- 1 Gajikwad Komal
- 2 Gajikwad Supriya
- 3 Ghunke Maykon
- 4 Gurav Prathmesh

Marks Obtained	Dated signature of Teacher
Process Related(35) Product Related(15) Total(50)	

I Practical Significance:

The arrays helps to represent collection of similar type of data under common name.

II Relevant Program Outcomes (POs)

- o Basic knowledge: Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- o Discipline knowledge: Apply Computer engineering discipline - specific knowledge to solve core computer engineering related problems.
- o Experiments and practice: Plan to perform experiments and practices to use the results to solve broad-based Computer engineering problems.
- o Engineering tools: Apply relevant Computer technologies and tools with an understanding of the limitations.
- o Communication: Communicate effectively in oral and written form.

III Competency and Practical skills

This practical is expected to develop the following skills in you :

Develop C++ programs to solve broad-based problems

1. Define one dimensional array.
2. Compile the program.
3. Debug and execute the program.

IV Relevant Course Outcome(s)

Develop C++ programs to solve problems using Procedure Oriented Approach..

V Practical Outcome (POs)

Practical Outcome (POs) / Execute simple C++ program using one dimensional Write/ Compile/ debug arrays.

VI Relevant Affective domain related Outcome(s)

Relevant Affective domain related Outcome(s) in C++.

1. Select proper programming environment in C++.
2. Follow ethical practices.

VII Minimum Theoretical Background

Arrays in C++

Arrays is a collection of data of same types stored in sequential memory location.

It is a linear data structure, where data is stored sequentially one after the other.

The elements in an array is accessed using an index.

For example, In an array of n elements, the first element has index zero and the last element has index (n-1).

Elements with consecutive index (i.e. 1 and i+1) are stored in consecutive memory location in the system.

Array can be divided into following types:

One Dimensional Array

Multi-Dimensional Array

One Dimensional Array Syntax:

type arrayName[arraySize];

Practical No. 4: Program to Implement One Dimension Array

I Practical Significance:
The arrays helps to represent collection of similar type of data under common name.

Relevant Program Outcomes (POs)

- Basic knowledge: Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- Discipline knowledge: Apply Computer engineering discipline - specific knowledge to solve core computer engineering related problems.
- Experiments and practice: Plan to perform experiments and practices to use the results to solve broad-based Computer engineering problems.
- Engineering tools: Apply relevant Computer technologies and tools with an understanding of the limitations.
- Communication: Communicate effectively in oral and written form.

XV Assessment Scheme

Performance indicators	Weightage
Process related(35 Marks)	70%
1 Logic formation	20%
2 Appropriate use of control structure (for, while, do-while)	20%
3 Debugging ability	10%
4 Follow ethical practices,	30%
Product related (15 Marks)	10%
5 Expected Output	10%
6 Timely Submission of report	10%
7 Answer to sample questions	100%
Total (50 Marks)	

List of Students Team Members

- Gaikwad Komal
- Gaikwad Supriya
- Ghunke Mukund
- Gupta Prathmesh

www.msbtresolution.xyz

Competency and Practical skills
This practical is expected to develop the following skills in you :

Develop C++ programs to solve broad-based problems

- Define one dimensional array.
- Compile the program.
- Debug and execute the program.

Relevant Course Outcome(s)

Develop C++ programs to solve problems using Procedure Oriented Approach..

Practical Outcome (POs)

Write/ Compile/ debug / Execute simple C++ program using one dimensional arrays.

Relevant Affective domain related Outcome(s)

Relevant Affective domain related Outcome(s) in C++.

- Select proper programming environment in C++.
- Follow ethical practices.

Minimum Theoretical Background

VII Arrays In C++

Arrays in C++
Array is a collection of data of same types stored in sequential memory location. It is a linear data structure, where data is stored sequentially one after the other. The elements in an array is accessed using an index.

For example, In an array of n elements, the first element has index zero

and the last element has index (n-1). Elements with consecutive index (i.e. 1 and i+1) are stored in consecutive memory location in the system.

Array can be divided into following types:

- One-Dimensional Array
- Multi-Dimensional Array

One Dimensional Array Syntax:
type arrayName [arraySize];

Marks Obtained	Dated signature of Teacher
35	13 46

Examples:

```
double salary[15000];
int age[5]={22,25,30,32,35};
```

VIII Resources required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware: Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows /LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- Handle computer system and peripherals with care.
- Follow safety practices.

X Resources used

S. No.	Name of Resource	Specification
1	Computer System with broad specifications	hp 230 G2 , i5 processor
2	Software	Turbo C++
3	Any other resource used	Google chrome

www.msbtresolution.xyz

```
float num[100], sum=0.0, average;
cout << "Enter the numbers of data: ";
cin >> n;
while (n > 100 || n <= 0)
{
    cout << "Error! number should in range of (1 to 100)." <<
        endl;
    cout << "Enter the number again: ";
    cin >> n;
}
for(i = 0; i < n; ++i)
{
    cout << "1 + 1 << ". Enter number: ";
    cin >> num[i];
    sum += num[i];
}
average = sum / n;
cout << "Average = " << average;
return 0;
```

- Which of the following correctly declares an array?
 a) int array[10]; b) int array;
 c) array[10]; d) array array[10];
- Can we change the size of an array at run time?
- What is the default value of Array?
- Can we declare array size as a negative number?
- What is the meaning of anonymous array? Explain with an example?
- Is there any difference between int[] a and int a[]?
- How to copy an array into another array?
- Can we change the size of an array at run time?
- Can you declare an array without assigning the size of an array?

(Space for Answers)

XI Result (Output of the Program)

~~we successfully executed the program to implement one directional array.~~

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- State output of the following code:

```
#include <iostream.h>
int main()
{
    int n, i;
```

- 1) `int arr[10];`
- 2) We can't change size of array at run-time
- 3) If no size is given to array, & if it is declared in local function, then it is unknown & take garbage value. If it is declared globally or static, then it will initialized to 0 by default if no values are provided.
- 4) If no size is given to array, & if it is declared in local function, then it is unknown & take garbage value. If it is declared globally or static, then it will initialized to 0 by default if no values are provided.
- 5) It is depend on what is actual location the memory. It may valid or invalid & it can be truncated.
- 6) Anonymus array is concept of Java. It is a array without name. It is an array just for creating & using instantly.

Q-7 Yes. there is difference between `int[] a` & `int a[]` because `int[]` makes the array to variable int but unfortunately '`int`' is a keyword & we could not use it as variable. Where as `int a[]` is correct declaration for array.

Q-8 Just put them in a loop. The most popular loop you can use & just use assignment separate to assign the value from RHS variable to LHS, & use another array name

```
array_name[i] = b[i];
```

No, we can't change array size at run time as it fixed at the time of declaration but you can change size but condition is that changed size & value should be less than actual size of array. You may change the size using another variable.

Q-9 You can't keep the size of array empty but you can declare array without size as follow -

```
int a // declare a  
cin >> a; // input
```

```
// assign a to size of array  
that's it.
```

XIII Exercise

Attempt any 2 programs from 1-4 and Q.5 attempt any 2 from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program to find median of two sorted arrays of same size.
2. Write a C++ program to find the two repeating elements in a given array.
3. Find the smallest and second smallest elements in an array.
4. Write a C++ program to find the Missing Number.
5. Complete the given table:

Name: Gaikwad Supriya Yuvraj

Roll no.: 10

Class: SYCSE.

Source Code

```
int getMedian(int ar1[], int ar2[], int n)
{
    int i = 0;
    int j = 0;
    int count;
    int m1 = -1, m2 = -1;
    for (count = 0; count <= n; count++)
    {
        if (i == n)
        {
            m1 = m2;
            m2 = ar2[0];
            break;
        }
        else-if (j == n)
        {
            m1 = m2;
            m2 = ar1[0];
            break;
        }
    }
}
```

```

if(n1 == n2)
{
    cout<<"Median is %d", getMedian(ar1, ar2, n1)<<endl;
}
else
{
    cout<<" Doesn't work for arrays of unequal size "<<endl;
}

int main()
{
    int ar1[] = {1, 12, 15, 26, 38};
    int ar2[] = {2, 13, 17, 30, 45};

    int n1 = sizeof(ar1)/sizeof(ar1[0]);
    int n2 = sizeof(ar2)/sizeof(ar2[0]);
}

```

www.msbtresolution.xyz

Output-----

⑥ ~~Output~~

Median is 16

```
/* smallest & second smallest element */
#include <iostream.h>
#include <stdio.h>
#include <conio.h>

void main()
{
    int a[30], n, i;
    cout << " Enter the no of elements ";
    cin >> n;
    cout << endl << " Enter array elements ";
    for (i=0; i<n; i++)
    {
        cin >> a[i];
    }
    bubbleSort(a, n);
    cout << " Smaller element = " << a[0] << endl;
    cout << " Second smallest = " << a[1] << endl;
    getch();
}

void bubbleSort(int arr[], int n)
{
    int i, j, temp;
    for (i=0; i<n-1; i++)
    {
        for (j=0; j<n-1-i; j++)
        {
            if (arr[j] > arr[j+1])
            {
                temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;
            }
        }
    }
}
```

* missing number in
* include < iostream.h
* include < conio.h
* include < stdio.h
* include < stdio.h

void main()

```
int n, a[10], total,
```

```
choice();
```

choice - enter the no of elements

```
cout << "enter no of array elements : " << endl;
```

```
cin >> n;
```

```
cin >> a[0];
```

```
total = (n+1) * (n+2) / 2,
```

```
tot = (n+1) * (n+2) / 2,
```

```
total = a[0];
```

```
count <= total <= end,
```

```
getch();
```

1 2 3 4 5 6

Output -
enter the no of elements 5
enter an array element 1 2 3 4 6
0 4

Program Code

```

#include <iostream.h>
int array1[] = {1200, 200, 200, 1230, 1342};
int array2[] = {12, 14, 16, 18, 20};
int temp, result, i, j;
void main()
{
    for (temp = 0; temp < 5; temp++)
    {
        result += array1[temp];
        for (temp = 0; temp < 5; temp++)
            result += array2[temp];
        cout << result;
    }
}

#include <iostream.h>
void main()
{
    int array1[] = {6, 2, 4, 6, 7, 5, 3};
    int n, result = 0;
    for (i = 0; i < n; i++)
    {
        result += array1[i];
    }
    cout << result;
}

#include <iostream.h>
void fun()
{
    char str[15] = "ABCDE";
    cout << str[3];
    cout << str;
}

(Space for answers)

```

(Space for answers)

```

void main()
{
    int a[10], i, j;
    clrscr();
    for (i = 0, j = 10; i != j)
    {
        cin >> a[i];
        if (i == 6, j < 9, i++)
        {
            for (j = i + 1; j < 10, j++)
            {
                if (a[i] == a[j])
                {
                    cout << "a[i]" << " is repeating element ";
                    getch();
                }
            }
        }
    }
}

```

Write & justify**Output**

First for loop will count result of elements in which 5 elements are present ad basic result will count same in original result but for element which added due to condition i.e. CID = 6+3 stored in array linearly, so in for statement loop, the addition of elements will replace.

CID = ABC because ABC is stored respectively in str[0] to str[4] & str[5] & str[6] & str[7] & str[8] & str[9] & str[10] & str[11] using last statement will occur.

For other s follows:

1. for (i = 0, j = 10, i != j)

2. if (i == 6, j < 9, i++)

3. for (j = i + 1, j < 10, j++)

4. if (a[i] == a[j])

5. cout << "a[i]" << " is repeating element "

6. getch();

VIII Resources required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware: Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows / LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- Handle computer system and peripherals with care.
- Follow safety practices.

X Resources used

S. No.	Name of Resource	Specification
1	Computer System with broad specifications	hp 280 G2 , is processor
2	Software	Turbo C++
3	Any other resource used	Google chrome

XI Result (Output of the Program)

We successfully executed a program to perform matrix operation using multidimensional array.

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- What is the error in this C++ code?

```
#include<iostream.h>
void main()
{
    int arr[2][3];
    arr[1][1] = {1, 2, 3}, {4, 5, 6};
    cout << arr[1][0];
}
```

- State output of the following code:

```
#include<iostream.h>
void main()
{
    // initializing the 3-dimensional
    array int x[2][3][2] =
    {
        {
            {
                {0,1}, {2,3}, {4,5} },
                {6,7}, {8,9}, {10,11}
            }
        };
    // output each element's value for
    // output each element's value for
    for (int i = 0; i < 2; ++i)
    {
        for (int j = 0; j < 3; ++j)
        {
            for (int k = 0; k < 2; ++k)
            {
                cout << "Element at x[" << i << "][" << j
                << "][" << k << "] = " << x(i)(j)(k)
                <<
            }
        }
    }
}
```

(Space for answers)

- How many kinds of elements an array can have?
 - Char and int type
 - Only char type
 - Only int type
 - All of them have same type

1) ERRORS –
1) Unable to open include file 'Iostream'.
2) Expression syntax
3) Undeclared symbol cout

1) Output –
element at &x[1][0][0]=0
element at &x[1][0][1]=1
element at &x[1][1][0]=2
element at &x[1][1][1]=3
element at &x[1][2][0]=4
element at &x[1][2][1]=5
element at &x[1][2][2]=6
element at &x[1][0][0]=7

element at $x[1][1][0] = 8$
 element at $x[1][1][1] = 9$

element at $x[1][2][0] = 10$
 element at $x[1][2][1] = 1$

3) \rightarrow All of them have same type

XIII. Exercise

Attempt Q1 or Q2 and Q3 a or b from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program to Multiply Two 3×3 Matrix Using Multi-dimensional Array
2. Write a C++ program to find Transpose of a 2×2 Matrix
3. Complete the given table:

Program Code	Write & justify Output
a) #include <iostream.h> void main() { int a[2][3] = {1, 2, 3, , 4, 5}; int i = 0, j = 0; for (i = 0; i < 2; i++) for (j = 0; j < 3; j++) cout<<a[i][j]; } b) #include <iostream.h> void main() { int a[2][3] = {1, 2, 3, 4, 5}; int i = 0, j = 0; for (i = 0; i < 2; i++) for (j = 0; j < 3; j++) cout<<a[i][j]; }	error - declaration is 123450

Name: Supriya Yuvraj Gaikwad.
 Roll no.: 10
 Class: SYCSE.

Source Code-----

void main()

```
int a[3][3], b[3][3], c[3][3], i, j, k, sum;
```

```
clrscr();
```

cout<<"\n Enter values for first 3×3 matrix:\n";

```
for (i = 0; i <= 2; i++)
```

```
{
```

```
    for (j = 0; j <= 2; j++)
```

```
        cin>>a[i][j];
```

```
}
```

cout<<"\n Enter values for second 3×3 matrix:\n";

```
for (i = 0; i <= 2; i++)
```

```
{
```

```
    for (j = 0; j <= 2; j++)
```

```
        cin>>b[i][j];
```

```
}
```

```
    for (i = 0; i <= 2; i++)
```

```
{
```

```
    for (j = 0; j <= 2; j++)
```

```
{
```

```
    C[i][j] = 0;
```

```
for ( k = 0 ; k <=2 ; k++ )  
    c[i][j] = c[i][j] + a[i][k] * b[k][j];
```

Enter values for second 3×3 matrix:

```
is: 0^n;
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

```
cout << "In Enter values for 2*2 matrix ";
for (i=0 ; i<2 ; i++)
{
    for (j=0 ; j<2 ; j++)
        cin >> a[i][j];
}

for (i=0 ; i<2 ; i++)
{
    for (j=0 ; j<2 ; j++)
        b[i][j] = a[i][j];
}

cout << "The transpose of the above matrix is";
for (i=0 ; i<2 ; i++)
{
    for (j=0 ; j<2 ; j++)
        cout << " " << b[j][i];
    cout << endl;
}
getch();

```

References / Suggestions for further Reading

- 1 <https://www.sanfoundry.com/multiple-choice-questions-c-multidimensional-arrays/>
- 2 <https://www.hackerearth.com/practice/data-structures/arrays/multi-dimensional/tutorial/>
- 3 <https://www.programiz.com/cpp-programming/examples/matrix-transpose>

Declaring Objects: When a class is defined, only the specification for the object is defined, no memory or storage is allocated. To use the data and access functions defined in the class, you need to create objects.

Syntax:
class Name ObjectName;

e.g.
Student std;

Accessing data members and member functions: The data members of class can be accessed using the dot(.) operator with member functions of class.

For example if the name of object is std and you want to access the member function with the name getData() then you will have to write std.getData().

Member Functions in Classes: There are 2 ways to define a member function:

Inside class definition:

Syntax: class className

```
// other members declaration
return type functionName(list of parameters)
{
    // body of function
}
// other members declaration
```

Outside class definition:

Syntax: return type className :: functionName(list of parameters)

```
// body of function
```

VII. Resources required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware: Computer System	Computer (i3-5th preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	# define batch size
2	Operating system	Windows / LINUX	For all Experiments	
3	Software	Turbo C++ Version 3.0 or any other	As per batch size	

IX. Precautions

- Handle computer system and peripherals with care.
- Follow safety practices.

X. Resources used

S. No.	Name of Resource	Specification
1	Computer System with broad specifications	Intel i3 - 3220 CPU, 3.30 GHz, 64 bit windows 7 Turbo C++
2	Software	
3	Any other resource used	Internet Software

XI. Result (Output of the Program)

(We successfully executed a program to implement class & object.)

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO. (Note: Use Point VII to X and XII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- What is the difference between struct and class in C++?
- State output of the following code:

```
#include<iostream.h>
class Empty {
public:
    void main()
    {
        cout << sizeof(Empty);
    }
}
```

- non-zero value b) 0 c) Compile time error d) Runtime error

- State True or False: Data items in C++ class are by default public. (Space for answers)

VIII. Resources required

2	# include <iostream.h>
3	# include <conio.h>
4	CLASS EMPTY {
5	void main()
6	{
7	cout << sizeof(EMPTY);
8	}
9	}

→ OUTPUT ⇒ 11

1) ~~Struct~~

2) ~~False~~

3) ~~True~~

- 1) ~~Struct~~
 1) Struct is a collection of variable & function
 of variable & different-language
 2) Function are not enclosed
 In structure definition

XIII Exercise

Attempt Q1 or Q2 and Q3 a or b from the following:
 (Note: Use Point VIII to X and XIII to XV for all relevant programming exercise like blank pages provided or attach more pages if needed.)

1. Write a C++ program to declare a class "Book" having data members book_name, author and price. Accept and display data for book having maximum price.
2. Write a C++ program to declare a class "Staff" having data members name, basic salary, DA, HRA and calculate gross salary. Accept and display data for one staff.
 - a. Where DA=74.5% of basic
 i. HRA=30% of basic.
 ii. Gross_salary=basic+HRA+DA
3. Complete the given table:

Program Code	Write & Justify Output
<pre>a) #include<iostream.h> class Empty {}; void main() Empty a, b; if (a == b) cout << "impossible" << endl; else cout << "Fine" << endl;</pre>	<p>Error- Illegal Structure operation.</p>

```
#include <iostream.h>
#include <stdio.h>
class book
{
public:
    char book_name[30];
    char author[20];
    int price;
    void get()
    {
        cout << " enter book name ";
        cin >> book_name[30];
        cout << " Enter author name ";
        .cin >> author[20];
        cout << " Enter book price ";
        cin >> price;
    }
    void display()
    {
        if (price > 300)
            cout << " Book name = " << book_name;
        cout << " Author name = " << author;
        cout << " Book price = " << price;
    }
};
```

Name-Gaikwad Supriya Yuvraj

Roll no.- 10

Output :-
enter book name - let us c '
enter author name - Balguru Swami'
enter book price - 250

```

/*Calculate gross salary of given basic salary*/
class staff
{
    float b_sal,g_sal;
    float hra,da;
public:
void enter()
{
    cout<<"Enter your basic salary-";
    cin>>b_sal;
}
void display()
{
    hra=(30+b_sal)100;
    da=(74.5+b_sal)/100;
    g_sal=b_sal+hra+da;
    cout<<"\n Gross salary is-<<g_sal;
}
};
```

```
b) #include <iostream.h>
class Rectangle {
    int width, height;
public:
    void set_values (int,int);
    int area();
    int width*height;
}
void main()
{
    staff s;
    s.set();
    s.display();
    getch();
}
```

Q
Gali

Output-

Enter your basic salary-25000
Gross salary is-5070.5

```
b) #include <iostream.h>
class Rectangle {
    int width, height;
public:
    void set_values (int,int);
    int area();
    int width*height;
}
void main () {
    rectangle rect;
    rect.set_values (3,4);
    cout << "area: " <<
    rect.area();
}
```

Output :-
Area =12
Justification =
we have to remove
return statement
in int area () function
because function
can return a
value but class can't
return a value rather
than object.

(Space for answers)

- V **References / Suggestions for further Reading**
1. <https://www.geeksforgeeks.org/c-classes-and-objects/>
 2. https://www.tutorialspoint.com/cplusplus/cpp_classes_objects.htm

Accessing data members and member functions: The data members and member functions of class can be accessed using the dot(.) operator with the object. For example if the name of object is std[2] and you want to access the member function with the name *getdata()* then you will have to write *std[2].getdata()*.

VIII Resources required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware: Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows /LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- Handle computer system and peripherals with care.
- Follow safety practices.

X Resources used

S. No.	Name of Resource	Specification
1	Computer System with broad specifications	Intel (i3) core (TM) i3 RAM - 4GB
2	Software	Turbo C++
3	Any other resource used	Internet penke?

(Space for answers)

```

char title [30];
float price ;
public:
void getdata () ;
void putdata () ;

void books :: getdata ()
{
cout<<"Title:"; cin>>title;
cout<<"Price:"; cin>>price;
}

void books :: putdata ()
{
cout<<"Title:"<<title<< endl;
cout<<"Price:"<<price<< endl;
}

const int size=3;

void main ()
{
books book[size];
for(int i=0;i<size;i++)
{
cout<<"Enter details of book "<<(i+1)<<"\n";
book[i].getdata();
}

for(int i=0;i<size;i++)
{
cout<<"\nBook "<<(i+1)<<"\n";
book[i].putdata();
}
}

```

XI Result (Output of the Program)

We understand + wrote program to implement
array of objects.

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- State the output of the following program:

```
#include<iostream.h>
class books
{

```

XIII. Exercise**Attempt Q1 or Q2 and Q3 a or b from the following:**

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise blank pages provided or attach more pages if needed.)

- Write a C++ program to define a class "Employee" having data members emp_id, emp_name and emp_salary. Accept and display data for employees having salary greater than 25,000/-.
- Write a C++ program to define a class "City" having data members name, population. Accept and display data for 10 cities.

3. Complete the given table:

Program Code

```
a) #include <iostream.h>
```

Write & justify Output
Output - Student1 Name: John Marks: 80

(Space for answers)

```

1) #include <iostream.h>
#ifndef include <conio.h>
class employee
{
    int emp_id , emp_sali
    char emp_name[10];
public :
    void accept()
    {
        cout << "Enter your id , name & Salary";
        cin >> emp_id >> emp_name >> emp_sali;
    }
    void display()
    {
        if (emp_sali > 25000)
    }
    cout << "Id = "<< emp_id << "\n Name = "<< emp_name << "\n Salary = "<< emp_sali;
}
void main()
{
    employee s;
    int i;
    clrscr();
    for (i=0;i<5;i++)
    {
        s.accept();
        s.display();
    }
    getch();
}

```

Name: Gaikwad Supriya Yuvraj

Roll no:10

C++ Program code:

#include<iostream.h>
#include<conio.h>

class city
{
 char name[20];
 long population;
public:
 void get()
 {
 cout << "Enter city name and population:";
 cin >> name >> population;
 }
}

```

www.msbtresolution.xyz
void main()
{
    void put()
    {
        cout << "City name: "<< name << "Population: "<< population;
    }
}

```

References / Suggestions for further Reading

1. <https://www.geeksforgeeks.org/c-classes-and-objects/>
2. https://www.tutorialspoint.com/cplusplus/cpp_classes_objects.htm

Assessment Scheme

Performance indicators	Weightage
Process related(35 Marks)	70%
1 Logic formation	20%
2 Appropriate class and array of objects definition	20%
3 Debugging ability	20%
4 Follow ethical practices.	10%
Product related (15 Marks)	30%
5 Expected Output	10%
6 Timely Submission of report	10%
7 Answer to sample questions	10%
Total (50 Marks)	100%

List of Students Team Members

1. Gajikwad Komal
2. Galkwad Supriya
3. Ghunke Maykan
4. Gumav Prathmesh

Marks Obtained	Dated signature of Teacher
Process Related(35) 33	Product Related(15) 13 Total(50) 46

```

city c[10];
int i;
clrscr();
for(i=0;i<10;i++)
{
    c[i].get();
    c[i].put();
}
getch(); ②
}

Output:
enter city name and population:mumbai 250000
city name:mumbai
population:250000
enter city name and population:pune 200000
city name:pune
population:200000
enter city name and population:sangli 100000
city name:sangli
population:100000
enter city name and population:nagpur 150000
city name:nagpur
population:150000
enter city name and population:ichalkaranji 100000
city name:ichalkaranji
population:100000
enter city name and population:

```

Practical No. 8: Program to Implement Friend Function

Practical

Friend return type friend function (name list of parameters).

Another member declaration

- I Practical Significance:** Classes and objects help to represent real life entity with different attributes. friend member functions to access the private or related member functions.
- The use of friend function allows access to member functions of another class.

II Relevant Program Outcome (POs)

- Basic knowledge:** Apply knowledge of basic mathematics, sciences and engineering to solve the broad-based computer engineering problems.
- Discipline knowledge:** Apply Computer Engineering discipline specific knowledge to solve core computer engineering problems.
- Experiments and practice:** Past to perform experiments and practices to use it results to solve technical computer engineering problems.
- Engineering tools:** Apply relevant Computer engineering tools with a understanding of the limitations.
- Communication:** Communicate effectively in oral and written form.

III Competency and Practical skills

This practical is expected to develop the following skills in you:

Develop C++ programs to solve broad based problems

- Define real life entity into classes and objects.
- Define and use of friend function.
- Compile the program.
- Debug and execute the program.

IV Relevant Course Outcomes

Develop C++ programs using classes and objects.

V Practical Outcome (POs)

Name: [Complete] defining / Execute simple C++ programs using classes, objects and friend function.

VI Refer and Attainable domain related Outcomes (KO)

- Select proper programming environment in C++.
- Follow safety measures.
- Follow ethical practices.

VII Minimum Theoretical Background

- Friend Function:** A friend function of a class is defined outside that class scope but it has the right to access all private and protected members of a class. Even though the prototypes for friend functions appear in the class function as friend of class.
- Declaring friend function

Syntax:
class className
{
 // other members declaration

VIII Resources required

Sl. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware Computer (1.1.5 preferable), RAM minimum 2 GB and monitor, HDD 40GB and above	Computer (1.1.5 preferable), RAM minimum 2 GB and monitor, HDD 40GB and above	As per batch size	For all Experiments
2	Software Turbo C++ Version 3.0 or any other	Windows / LINUX		

IX Presentations

- Handle computer system and peripherals with care.
- Follow safety practices.

X Resources used

Sl. No.	Name of Resource	Specification
1	Computer System with broad specifications	INTEL (P-) CORE (TM) i3 2.8GHz
2	Software	Turbo C++
3	Any other resource used	Printer

XI Result (Output of the Program)

We understood & wrote program to implement friend function.

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of Identified CO.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Can member functions of one class be friend functions of another class?
 a) Yes b) No

2. A function can be declared as friend maximum only in two classes.
 a) True b) False

3. Which of the following rules will not affect the friend function?
 a) private and protected members of a class cannot be accessed from outside
 b) private and protected member can be accessed anywhere

- c) protected member
 d) none of the mentioned

- d) none of the keyword 'friend' should be placed?

4. Where does keyword declaration
 a) function declaration
 b) function definition
 c) main function
 d) none of the mentioned

(Space for answers)

Program Code	Write & justify Output
<pre>#include <iostream.h> class A { int a; public: A() {a = 10;} friend void showA(A& x); }; void showA(A& x) { cout << "a=" << x.a; } void main() { A a; showA(a); }</pre>	<p>1) a=10</p>

Program Code	Write & justify Output
<pre>#include <iostream.h> class Box { double width; public: friend void PrintWidth(Box box); void setWidth(double wid); }; void Box::setWidth(double wid) { width = wid; } void PrintWidth(Box box) { box.width = box.width * 2; cout << "Width of box : " << box.width << endl; } void main() { Box box; box.setWidth(10.0); PrintWidth(box); }</pre>	<p>2) width of box : 20</p>

XIII Exercise

Attempt Q1 or Q2 and Q3 a or b from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program to swap the values of two variables using friend function.

2. Write a C++ program to define a class "Distance" having data member as meters.

- Display the addition of two Distance objects using friend function.

3. Complete the given table:

```
1) #include <iostream.h>
   #include <conio.h>
class swap
{
    int no1 , no2 , kmp;
public:
    void accept()
    {
        cout << "Enter any two numbers in ";
        cin >> no1 >> no2;
    }
    friend swap (swap s)
    {
        int temp;
        swap s;
        s.temp = s.no1;
        s.no1 = s.no2;
        s.no2 = temp;
        cout << "Swapped values are ";
        cout << s.no1 << ", " << s.no2;
    }
};

void main()
{
    swap s1;
    cout << "Enter first number : ";
    cin >> s1;
    cout << "Enter second number : ";
    cin >> s2;
    cout << "Swapped Values are : ";
    cout << s1;
}
```

Name: Galkwad Supriya Yuvraj.

Roll No.: 10.

#include<stdio.h>

#include<conio.h>

#include<iostream.h>

class distance

int m;

public:

void enter()
{
 cout << "Enter any distance in meter";
 cin >> m;
}

void calculate (distance , distance);
}
};

void calculate (distance d1, distance d2)
{
 cout << "Addition of two distance: " << m << d1.m + d2.m << "\n";
}

```
void main()
{
    distance d1;
    distance d2;
    cout << "Enter first distance : ";
    cin >> d1;
    cout << "Enter second distance : ";
    cin >> d2;
    cout << "Sum of two distances : ";
    cout << d1 + d2;
}
```

References / Suggestions for further Reading

- 1 <https://www.geeksforgeeks.org/c-classes-and-objects/>
- 2 https://www.tutorialspoint.com/cplusplus/cpp_classes_objects.htm
- 3 <http://www.sanfoundry.com/multiple-choice-questions-c-plus-plus-friends/>

Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		
1 Logic formation		70%
2 Appropriate friend function declaration and definition		20%
3 Debugging ability		20%
4 Follow ethical practices.		10%
Product related (15 Marks)		30%
5 Expected Output		10%
6 Timely Submission of report		10%
7 Answer to sample questions		10%
Total (50 Marks)		100%

List of Students / Team Members

1. Gopal kumar komal.....
 2. Gopal kumar Supriya.....
 3. Ghunake Muskhan.....
 4. Gurav Prathmesh

Marks Obtained		Dated signature of Teacher	
Process Related(35)	Product Related(15)	Total(50)	
93	12	45	<u>Op</u>

Practical No. 9: Program to Implement Inline Function

Syntax:

```
class className
{
    //other members declaration public:
        inline return type FunctionName(list of parameters)
    {
        //body of inline function
    }
};
```

- Practical Significance:** help to represent real life entity with different attributes
The classes and objects. facilitates faster execution of the program.

The use of inline functions (POs)

- II Relevant Program Outcomes (POs)**
- o Basic knowledge: Apply knowledge of basic mathematics, sciences and engineering to solve the broad-based Computer engineering problems.
 - o Discipline knowledge: Apply engineering discipline related problems.
 - o Knowledge to solve core computer engineering problems.
 - o Experiments and practice: Plan to perform experiments and practices to solve core engineering problems.
 - o Engineering tools: Apply relevant Computer technologies and tools results to solve broad-based Computer engineering problems.
 - o Understanding of the limitations.
 - o Communication: Communicate effectively in oral and written form.

- III Competency and Practical skills**
- This practical is expected to develop the following skills in you :

- Develop C++ programs to solve broad-based problems**
1. Define real life entity into classes and objects.
 2. Define and use of inline function.
 3. Compile the program.
 4. Debug and execute the program.

- IV Relevant Course Outcome(s)**
- Develop C++ programs using classes and objects.

- V Practical Outcome (POs)**
- Write/ Compile/ debug / Execute simple C++ program using classes, objects and inline functions.

- VI Relevant Affective domain related Outcome(s)**
1. Select proper programming environment in C++.
 2. Follow safety measures
 3. Follow ethical practices.

- VII Minimum Theoretical Background**
- Inline Function:** C++ inline function is concept that is commonly used with inline qualifier in case defined function is more than a line.
- A function definition in a class definition is an inline function definition, even the use of the **inline** keyword.

- To inline a function, place the keyword **inline** before the function name and use function before any calls are made to the function. The compiler can ignore inline qualifier in case defined function is more than a line.
- Declaring inline function:**

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Computer System with broad specifications	Intel (P) core (i3) RAM - 8GB	As per batch size	For all Experiments
2	Software	Turbo C++	any other	
3	Operating system	Windows / LINUX		

- X Precautions**
1. Handle computer system and peripherals with care.
 2. Follow safety practices.

Resources used

S. No.	Name of Resource	Specification
1	Computer System	Intel (P) core (i3) RAM - 8GB
2	Software	Turbo C++

XI Result (Output of the Program)

COD. Successfully executed program to implement inline function.

.....
.....
.....
.....
.....

- To inline a function, place the keyword **inline** before the function name and use function before any calls are made to the function. The compiler can ignore inline qualifier in case defined function is more than a line.
- Declaring inline function:**

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. What does the **inline** keyword do?
 - A. Indicates a function declaration only within the same source code file
 - B. Tells the compiler to use the function calls to be replaced by the code from the function
 - C. Causes all function declarations
 - D. Allows one-line function declarations
2. Why would you want to use **inline** functions?
 - A. To decrease the size of the resulting program
 - B. To increase the speed of the resulting program
 - C. To simplify the source code file
 - D. To remove unnecessary functions
3. Which of the following is a limit on **inline** functions?
 - A. **Inline** functions cannot return a value.
 - B. **Inline** functions must return a value.
 - C. **Inline** functions must be less than ten lines.
 - D. The compiler may choose to ignore an **inline** directive
4. Which of the following is a valid **inline** for function **foo**?
 - A. **inline void foo()**
 - B. **void foo() inline**
 - C. **inline;void foo()**
 - D. None of the above

(Space for answers)

→ a) Cause all function call to be replaced by the code from the function.

→ b) To increase the speed of the resulting program.

→ c) The compiler may choose to ignore an **inline** directive.

→ d) **inline void** too § 3

XIII

Exercise

Attempt Q1 and Q2 from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program to create a class "Number" having data members **x** and **y** and perform mathematical operations like addition, subtraction, multiplication and division on two numbers using **inline** functions.
2. Complete the given table:

Program Code	Write & justify Output
<pre>a) #include <iostream.h> class InlineDemo { public: int square(int s); // declare the function }; //use inline prefix inline int InlineDemo::square(int s) { return s*s; }</pre>	<p>Square of 100</p> <p>Q. No. 15:-</p>

(Space for answers)

```

class number
{
    float x, y;
public:
    public float mul (float a , float b)
    {
        return (a*b);
    }
    float div (float a , float b)
    {
        return (a/b);
    }
    float add (float a , float b)
    {
        return (a+b);
    }
    float sub (float a , float b)
    {
        return (a-b);
    }
    void getdata()
    {
        cout << "Enter value of x & y : ";
        cin >> x >> y;
    }
    void display()
    {
        cout << "Multiplication : " << mul(x,y);
        cout << " Division : " << div (x,y);
        cout << "n Addition : " << add (x,y);
        cout << "n Subtraction : " << sub (x,y);
    }
    void main()
    {
        number n;
        n.getdata();
        n.display();
    }
}

```

- XIV References / Suggestions for further Reading**
1. <https://www.geeksforgeeks.org/c-classes-and-objects/>
 2. https://www.tutorialspoint.com/cplusplus/cpp_classes_objects.htm
 3. <https://www.geeksforgeeks.org/inline-functions-cpp/>

Assessment Scheme	
Performance indicators	Weightage
Process related(35 Marks)	
1 Logic formation	20%
2 Appropriate inline function declaration and definition	20%
3 Debugging ability	10%
4 Follow ethical practices.	30%
Product related (15 Marks)	
5 Expected Output	10%
6 Timely Submission of report	10%
7 Answer to sample questions	100%
Total (50 Marks)	

List of Students Team Members

- 1 Gaitkood, Supriya
- 2 Gaitkood, Komal
- 3 Ghamake, Mukund
- 4 Gurav, Prathmesh

Marks Obtained	Dated signature of Teacher
33	12

Declaring destructor function:

```
Syntax:  
class className  
{  
    //other members  
    declaration public:  
    //other members declaration  
    -className()  
    {  
        //body of destructor  
        function  
    }  
};
```

- Note:** Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.
- (Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

VIII Resources required

Sr. No.	Name	Specification	Quantity	Remarks
1	Hardware: Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows /LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- Handle computer system and peripherals with care.
- Follow safety practices.

X Resources used

S. No.	Name of Resource	Specification
1	Computer System with broad specifications	Intel (P) Core (TM) i3 RAM -4GB
2	Software	Turbo C++
3	Any other resource used	None net pointed.

XI Result (Output of the Program)

We understand & wrote program for
Implementation constructor & destructor

XII

Practical Related Questions

- Note:** Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.
- (Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- What happens when a class with parameterized constructors and having no default constructor is used in a program and we create an object that needs a zero-argument constructor?
- Compile-time error. b. Preprocessing error. c. Runtime error. d. Runtime exception.
- For automatic objects, constructors and destructors are called each time the objects enter and leave scope
 - inherit parent
 - class c. are constructed
 - are destroyed
- Which of the following statement is correct about the program given below?


```
#include<iostream.h>
class abc
{
public:
    abc()
    {
        cout<<"Welcome";
    }
    ~abc()
    {
        cout<<"India";
    }
};

void main()
{
    abc obj;
}
```

- a. The program will print the output Welcome.
 b. The program will print the output India.
 c. The program will print the output Welcome India.
 d. The program will report compile time error.

(Space for answers)

3. Complete the given table:

Program Code	Write & Justify Output
<pre>#include <iostream.h> int val = 0; class Test { public: Test() { cout<< ++val; } ~Test() { cout<< val--; } }; void main() { Test obj1, obj2, obj3; Test obj4; }</pre> <p>b) #include <iostream.h> class construct { public: float area; // Constructor with no parameters construct() { area = 0; } construct(int a, int b) { area = a * b; } void disp() { cout<< area<< endl; }</p>	<pre>12349 1234 See the value of constructor only on 4 Get the value of destructor</pre> <p>6 200.</p>

XIII Exercise

Attempt Q1 and Q2 from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- ~1. Write a C++ program to define a class "Number" having data members x and y and perform mathematical operations like addition, subtraction, multiplication and division on two numbers using constructor and destructor functions.
- 2. Write a C++ program to define a class "Product" having data members Prod_id, Prod_name, Prod_price. Accept and display data for 3 products using constructor overloading.

(Space for answers)

Program Code	Write & justify Output
<pre>a) #include <iostream.h> int val = 0; class Test { public: Test() { cout << ++val; } ~Test() { cout << val--; } }; void main() { Test obj1, obj2, obj3; Test obj4; }</pre>	<p>12349 1234 give the value of constructor only on 4 give the value of destructor</p>

```
b) #include <iostream.h>
class construct
{
public:
    float area;
    // Constructor with no Parameters
    construct()
    {
        area = 0;
    }
    construct(int a, int b)
    {
        area = a * b;
    }
    void disp()
    {
        cout << area << endl;
    }
};

void main()
{
    // Constructor Overloading
    // with two different constructors
    // of class name
    construct o1;
    construct o2(10, 20);
    o1.disp();
    o2.disp();
}
```

200

XIII Exercise

Attempt Q1 and Q2 from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program to define a class "Number" having data members x and y and perform mathematical operations like addition, subtraction, multiplication and division on two numbers using constructor and destructor functions.
2. Write a C++ program to define a class "Product" having data members Prod_id, Prod_name, Prod_price. Accept and display data for 3 products using constructor overloading.

/*carry various mathematical operations using constructor & destructor*/

class number

```
#include <iostream.h>
class product
{
    int pro_id,
        pro_name [20];
    char pro_price;
    int pro_price;
public:
    product (int i, char s[], int p)
    {
        pro_id = i;
        strcpy (pro_name, s);
        pro_price = p;
    }
    void display()
    {
        cout << "product-id " << product_id;
        cout << "product-price " << endl;
        cout << "name of product price " << endl;
        cin >> pro_id >> pro_name >> pro_price;
    }
};

void main()
{
    class product ob1;
    ob1.add();
    ob1.display();
}
```

```
int a,b,r1,r2,r3,r4;
number(int x,int y)
{
    a=x;
    b=y;
}
```

void add()

{

r1=a+b;

cout << "/n Addition-><r1";

}

void sub()

{

r2=a-b;

cout << "/n Subtraction-><r2";

(Space for answer)

Name-Gaikwad Supriya Yuvraj.

Roll no.-10

3) #include <iostream.h>

class product

{ int pro_id,

char pro_name[20],

int pro_price;

public:

product (int i, char s[], int p)

{

pro_id=i;

strcpy (pro_name,s);

pro_price=p;

}

void display()

{ cout << "product_id" << Product_id pro

name & product price" << endl;

cin >> pro_id >> pro_name >> pro_price;

void add()

{

r1=a+b;

cout<<"/n Addition-><r1;"

}

Product t1(1,"book",5000);

Product t2;

t1.display();

t2.display();

getch();

}

cout<<"\n Subtraction-><t2;"

}

```

void mul()
{
    r3=a*b;
    cout<<"\n Multiplication-><r3>";

}

void div()
{
    r4=a/b;
    cout<<"\n Division-><r4>";

}

void number()
{
    cout<<"\n Number-><r5>";

}

void main()
{
    number n(5,4);
    n.add();
    n.sub();
    n.mul();
    0getch();
    n.div();
    getch();
}

```

Output-
 Addition-9
 Subtraction-1
 Multiplication-20
 Division-0

X **Precautions**

1. Handle computer system and peripherals with care.
2. Follow safety practices.

X **Resources used**

X Resources used	Specification
1 Computer System with broad specifications	Intel CPU 1 GHz RAM = 4 GB
2 Software	Turbo C++
3 Any other resource used	Internet Printer.

XI **Result (Output of the Program)**

Result of the program to implement
Multiple Inheritance
Single inheritance

XII **Practical Related Questions**

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

blank pages provided or attach more pages if needed.)

1. A derived class with only one base class is called _____ inheritance.
 - a) single
 - b) multiple
 - c) multilevel
 - d) hierarchical
2. When a base class is privately inherited by the derived class, then _____.
 - a. protected members of the base class become private members of derived class
 - b. public members of the base class become private members of derived class
3. The derivation of Child class from Base class is indicated by _____ symbol.
 - a. ::
 - b. .
 - c. ;
 - d. |

(Space for answers)

1) → a) Single

2) → c) Both a & b

3) → b) :

XIII **Exercise**

Attempt Q1 and Q.2 from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program to define a class "Marks" from "Student" having data members roll_no, name. Derive a class "Marks" from "Student" having data members m1,m2,m3, total and percentage. Accept and display data for one student.

2. Write a C++ program to define a class "Employee" having data members emp_no, emp_name and emp_designation. Derive a class "Salary" from "Employee" having data members basic, hra, da, gross_sal. Accept and display data for one employee.

Write & justify

Ques: Consider the given code.

Program Code

Output -

The program
executed

multiple
declaration
for
variables

class
Base
public:
Base(0.5)
-Base(0.5)
protected:
protected

class
Derived
public:
Base

protected:
protected

private:
private

int main()
{
cout << "The program executed" <<

endl;

3) #include<iostream.h>

Output:

```
class Test
{
    int value;
public:
    Test(int v = 0) { value = v; }
    int getValue() { return value; }
};

void main()
{
    const Test t;
    cout << t.getValue();
}
```

Name-Gaikwad Supriya Yuvaraj.

Roll no.- 10

/*Calculate gross salary of given basic salary using inheritance*/

class employee

multiple
declaration for
derived

constructor
is done so
by removing

If we go
above
Output

int e_no;

public:

void enter()

```
{
    cout<<"\n Enter ur name-";
    cin>>name;
    cout<<"\n Enter ur emp_no-";
    cin>>e_no;
```

```
cout<<"\n Enter ur designation-";
cout << designation;
```

class salary;public employee

void main()

protected:

float g_sal,b_sal;

float da,hra;

public:

void calculate()

{

cout<<"\n Enter ur basic salary:";

cin>>b_sal;

hra=(30+b_sal)/100;

da=(74.5+b_sal)/100;

✓ g_sal=b_sal+hra+da;

void display()

{

cout<<"\n Name-"< <<name;

cout<<"\n Employee number-"< <<e_no;

cout<<"\n Designation-"< <<designation;

cout<<"\n Gross salary-"< <<g_sal;

Name-Priya

Employee number-4532

Designation-Manager

Gross salary is-5070.5

salary a:

a.enter();

a.calculate();

a.display();

getch();

Output
Enter ur name-Priya
Enter ur emp_no-4532
Enter ur designation-Manager
Enter ur basic salary-25000

www.mshtesolution.xyz

```
#include <iostream>
using namespace std;

class student
{
    int roll-no;
    char name[20];
public:
    void enter()
    {
        cout << "Enter your name & roll-no";
        cin >> name >> roll-no;
    }
    void display()
    {
        cout << "Name = " << name;
        cout << " Roll No = " << roll-no;
    }
};

class marks : public student
{
    int m1, m2, m3, total;
    float per;
public:
    void enter2()
    {
        cout << "Enter marks of m1, m2 & m3: ";
        cin >> m1 >> m2 >> m3;
    }
};
```

(Space for answer)

```

void display2()
{
    total = m1+m2+m3;
    per = total / 300;
    cout << " Total = " << total;
    cout << " In percentage = " << per;
}

```

```

}
void main()
{

```

```

marks m;
m. enter1();
m. enter2();
m. display();
m. display2();
getch();
}

```

~~Output:~~
Enter your name & roll-no,

Riya

~~8~~
Enter your marks of m1, m2, m3 :

90, 95, 87

Name : Riya

Roll No : 8

Total = 272

Percentage = 90.66

References / Suggestions for further Reading

1. <https://www.geeksforgeeks.org/inheritance-in-c/>
2. <https://www.geeksforgeeks.org/single-inheritance/>
3. <http://www.siteforinfotech.com/2017/05/top-20-mcq-questions-inheritance-in-cpp.html>
4. <https://www.careermile.com/mcq/inheritance-c-mcq-questions-and-answers-114.aspx>
5. <https://www.geeksforgeeks.org/output-of-c-program-set/>
6. <https://ide.geeksforgeeks.org/index.php>

Syntax of multilevel inheritance:

```

class base_classname
{
    properties;
    member functions;
};

class derived_classname1 : visibility_mode base_classname
{
    properties;
    member functions;
};

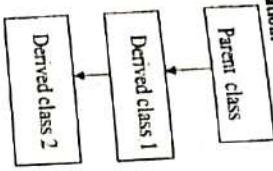
class derived_classname2 : visibility_mode derived_classname1
{
    properties;
    member functions;
};

```

```

class derived_classname2 : visibility_mode derived_classname1
{
    properties;
    member functions;
};

```

Pictorial Representation:**VIII Resources required**

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows / LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- 1 Handle computer system and peripherals with care.
- 2 Follow safety practices.

Resources used

S.No.	Name of Resource	Specification
1	Computer System with broad specifications	hp 280 G2 , i3 processor
2	Software	Turbo C++
3	Any other resource used	

XI Result (Output of the Program)

We learnt to execute simple C++ program using Multilevel Inheritance.

XII**Practical Related Questions**

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.
(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

```

1.
#include <iostream.h>
class A
{
public:
void print() { cout << "print() in A"; }
};

class B : private A
{
public:
void print() { cout << "print() in B"; }
};

class C : public B
{
public:
void print() { cout << "print() in C"; A::print(); }
};

void main()
{
    C b;
    b.print();
}

```

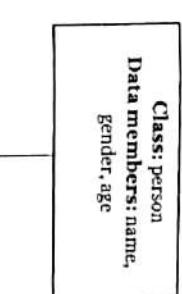
- a) print() in A
b) print() in B
c) Complete time error
d) None of the above

2. A class can be derived from another derived class which is known as
 inheritance.
3. In inheritance, the constructors are executed in the order
 of inheritance.
4. base class and derived class relationship comes under
 A) multipath B) multiple inheritance C) encapsulation D)None

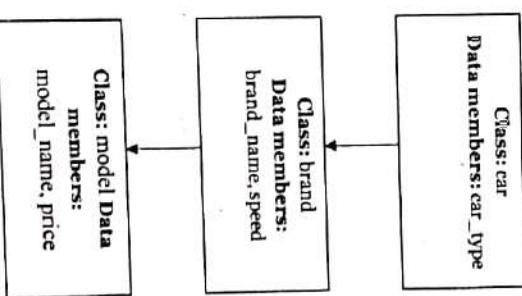
(Space for answers)

Q.1

- 1) Compile time error.
- 2) A class can be derived from another derived class which is known as multilevel inheritance.
- 3) In multilevel inheritance the constructors are executed in the order of inheritance.
- 4) Base class & derived class relationship comes under inheritance.



2. Write a C++ program to implement following Multilevel Inheritance



(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- I Write a C++ program to implement following Multilevel Inheritance

VIII Exercise
Attempt Q1 and Q.2 from the following:

```

1) #include <iostream.h>
# include < iostream.h >
class person
{
    char name [20];
    char gender [20];
    int age;
public:
    void get()
    {
        cout << "Enter the person name ,gender ";
        cin >> name >> gender >> age ;
    }
    void display ()
    {
        cout << name << "\n " << gender << "\n";
        cout << age;
    }
};

3) class employee : public person
{
    int emp_id;
    long salary;
    char company [20];
public:
    void set()
    {
        cout << " Enter the language known to
        programme ";
        cin >> no_of_lang_known;
    }
    void data()
    {
        cout << no_of_lang_known;
    }
};

IV References / Suggestions for further Reading
1. https://www.programmatically.net/cplusplus/docs/single-inheritance/
2. https://www.geeksforgeeks.org/inheritance-in-c/
3. http://www.siteforsofttech.com/2017/05/top-20-mcq-questions-inheritance-
   cpp.html
4. https://www.careerline.com/mcq/inheritance-c-mcq-questions-and-answer-
   114.aspx

```

Name : Supriya Yuvraj Gaikwad.

Roll No: 10

```
void main()
{
    programmee p;
```

```
    clrscr();
```

```
    p.get();
```

```
    p.display();
```

```
    p.ser();
```

```
    p.show();
```

```
    p.put();
```

```
    p.data();
```

```
    getch();
```

```
}
```

Output :-

Enter the person name , gender & age

projkt

Female

20

Enter the employee id , salary & company

1

50,000

TATA

enter the language known to programme

q.

Name : projkt

Age : 32

Gender : Female

Page No. _____
Date _____

```
char brand_name[20];  
int speed;  
public:  
void set()
```

```
- cout<<"enter the car brand name and speed";  
- cin>>brand_name>>speed;  
- void show()  
{  
    cout<<brand_name<<"\n"<<speed<<endl;
```

```
    cout<<"enter the car brand name and speed";  
    cin>>brand_name>>speed;
```

```
    void show()  
{  
    cout<<brand_name<<"\n"<<speed<<endl;
```

```
    cout<<brand_name<<"\n"<<speed<<endl;
```

```
    class model:public brand  
    {  
        char model_name[20];
```

```
        long price;
```

```
    public:  
        void put()
```

```
{  
    cout<<"enter the model name and price";
```

```
    cin>>model_name>>price;
```

```
    void data()  
{  
    cout<<model_name<<"\n"<<price;
```

```
    cout<<model_name<<"\n"<<price:
```

```
    void data()
```

Assessment Scheme

Performance Indicators	Weightage
Process related(35 Marks)	70%
Proper formation of code	20%
Appropriate base class and derived classes	20%
Declaration using multilevel inheritance	
Following ability	20%
Follow ethical practices.	10%
Product related (15 Marks)	30%
Expected Output	10%
Timely Submission of report	10%
Answer to sample questions	10%
Total (50 Marks)	100%

Students Team Members

Komal...Gatkhand...

Supriya...Gatkhand...

Muskan...Ghunake

Mathimah...Gurav...

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related(15)	Total(50)	
33	12	45	9/12/2016

Date

81

```

void<<model_name<<"\n"<<price;
cout<<endl;
cin>>model;
void main()
{
    model m;
    m.des();
    m.get();
    m.show();
    m.put();
    m.data();
    m.get();
    cout<<endl;
}

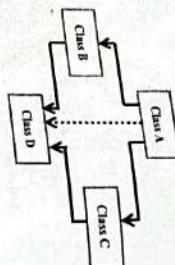
OUTPUT:
co Enter the car type: sedan
ci Sedan
Enter the brand name and speed audi 3000
co Audi 3000
Enter the model name and price Volvo 500000
co Volvo
500000

```

Virtual Base Class:

Virtual Base class: An ambiguity can arise when several paths exist to a class from the same base class. This means that a child class could have duplicate sets of members inherited from a single base class. When a class is derived regardless of the C++ solves this issue by introducing a virtual base class. When a class is made virtual, it ensures care is taken so that the duplication is avoided regardless of the number of paths that exist to the child class.

When two or more classes are derived from a common base class, we can prevent multiple copies of the base class being present in an object derived from those objects by declaring the base class as virtual. This can be achieved by preceding the base class's name with the word **virtual**.



Syntax:

class derived class_name virtual visibility mode base_class;

members of derived class;

VIII Resources required

Sr. No.	Name of Resource	Quantity	Remarks
1	Hardware: Computer System	1	
2	Software	1	
3	For all Experiments	1	

Precautions

1. Handle computer system and peripherals with care.

2. Follow safety practices.

Resources used

S. No.	Name of Resource	Specification
1	Computer System	HP 280 G2, i3 processor with broad specifications
2	Software	Turbo C++
3	Any other resource used	—

XI Result (Output of the Program)

We learn that to implement programming in C++ using concept of multiple inheritance.

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions, so as to refresh the achievement of identified CO.

(Note: Use Point VIII to X and All to XV for all relevant programming exercise use blank pages provided or attach new pages if needed.)

- What is the difference between multiple and multi-level inheritance in C++?
- State output the following code:

#include <iostream.h>

```

class liquid
{
public:
    float specific_gravity;
    void input()
    {
        cout << "specific gravity: ";
        cin >> specific_gravity;
    }
    void output()
    {
        cout << "specific gravity: " << specific_gravity;
    }
};

class fuel
{
public:
    float rate();
};

fuel::float rate()
{
    void input()
    {
        cout << "Rate(per liter): $";
        cin >> rate;
    }
    void output()
    {
        cout << "Rate(per liter): $" << rate;
    }
};
  
```

```

obj.c = 30;
obj.d = 40;
cout << "\n A : " << obj.a;
cout << "\n B : " << obj.b;
cout << "\n C : " << obj.c;
cout << "\n D : " << obj.d;

}

class Petrol: public liquid, public fuel
{
public:
    void input()
    {
        liquid::input();
        fuel::input();
    }

    void output()
    {
        liquid::output();
        fuel::output();
    }
};

int main()
{
    Petrol p;
    cout << "Enter data" << endl;
    p.input();
    cout << endl << "Displaying data" << endl;
    p.output();
    getch();
    return 0;
}

```

Multiple Inheritance

In this one class is derived from another class & that class derived from another class.

Multiple Inheritance

Specific gravity

$$g_0 \text{ (per litre)} = 3000$$

3.

- A - 100
B - 20
C - 30
D - 40

Q. c) Both dynamic binding & polymorphism

4. Which of the following advantages we lose by using multiple inheritance?
 a) Dynamic binding
 b) Polymorphism
 c) Both Dynamic binding & Polymorphism
 d) None of the mentioned

(Space for answers)

```

1. i) class petrol: public liquid, public fuel
   {
      public:
         void input()
         {
            liquid::input();
            fuel::input();
         }
      void output()
      {
         liquid::output();
         fuel::output();
      }
   }

2. i) int main()
   {
      petrol p;
      cout<<"Enter data" << endl;
      p.input();
      cout<<"Displaying data" << endl;
      p.output();
      getch();
      return 0;
   }

```

www.msbtosolution.xyz

4. Which of the following advantages we have by using multiple inheritance?
 a) Dynamic binding
 b) Polymorphism
 c) Both Dynamic binding & Polymorphism
 d) None of the mentioned

(Space for answer)

J. Multilevel

In this one class is derived from another class & that class derived from another.

Q. Output -
 Specific - gravity
 go
 rate (per litre)
 3000

3. A - 100
 B - 200
 C - 300
 D - 400

Q. c) Both dynamic bind

Dynamic

Bind

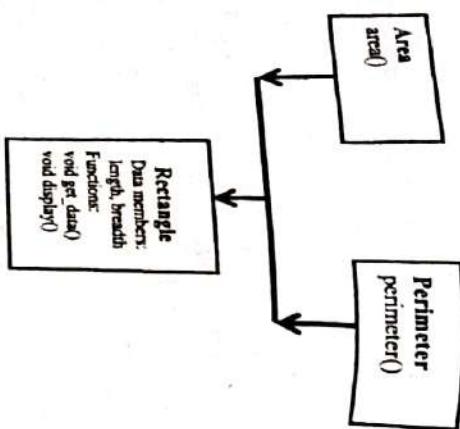
Obj

XIII Exercise
Attempt Q1 or Q2 and Q3 or b from the following:

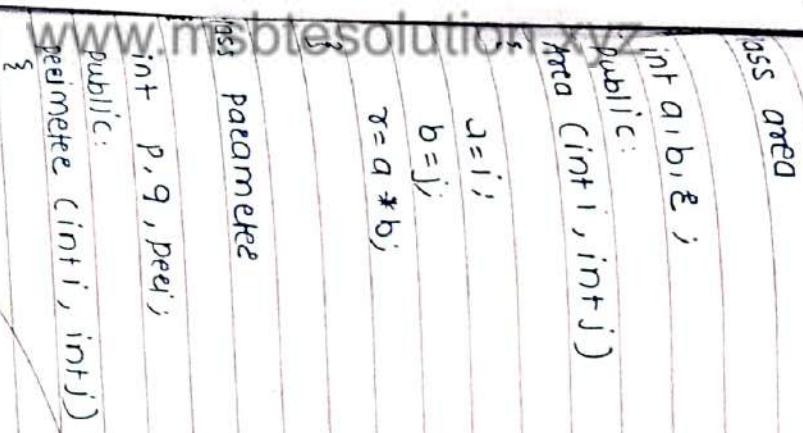
Note: Use Point VIII to X and XIII to XV for all relevant programming exercises.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercises.)
(Note: Use Point VIII to X and XIII to XV for all relevant programming exercises.)

1. Write a C++ program to calculate the area and perimeter of rectangles using concept of inheritance.



2. Write a C++ program for representation of class hierarchy as below. Assume



Pass Parameter

```
int p, q, per;
```

```
public:
```

```
perimeter (int i, int j)
```

```
{
```

```
p = i;
```

```
q = j;
```

```
per = 2 * (p + q);
```

```
}
```

```
class rectangle : public Area, public Perimeter
```

```
{
```

```
public:
```

```
rectangle (int a, int b, int c, int d)
```

```
area (int a, int b) : perimeter (int c, int d)
```

Name: Gaikwad Supriya Yuvraj.
Class: SY CSE.

Roll no.: 10.
/*a program for hybrid inheritance*/

```
#include<iostream.h>
class cricketer
{
    char c_name;
    int c_score;
public:
void enter1()
{
    cout<<"\n Enter cricketer's name and score: ";
    cin>>c_name>>c_score;
}
void display1()
{
    cout<<"\n Cricketer's name: "<<c_name;
    cout<<"\n Cricketer's score: "<<c_score;
}
};

class batsman:public virtual cricketer
{
    char b_name;
    int wickets;
public:
void enter2()
{
    cout<<"\n Enter bowler's name and no. of wickets taken: ";
    cin>>b_name>>wickets;
}
void display2()
{
    cout<<"\n Bowler's name: "<<b_name;
    cout<<"\n No. of wickets taken: "<<wickets;
}

class batsman:public virtual cricketer
{
    int runs;
    char bt_name;
}
```

Output:
Enter cricketer's name and score:
Sachin Tendulkar

public:
void enter3()
|
cout<<"\n Enter batsman's name and no. of runs made:";
cin>>bt_name>>runs;

void display3()

|
cout<<"\n Batsman's name:"<<bt_name;
cout<<"\n No. of runs made:"<<runs;

|

|
class allrounder{public bowler public batsman

|
int a_score;

char a_name;

public:

void enter4()

|
cout<<"\n Enter allrounder's name and ur score:";
cin>>a_name>>a_score;

|
void display4()

|
cout<<"\n Allrounder's name:"<<a_name;
cout<<"\n Allrounder's score:"<<a_score;

|

|
void main()

|
allrounder n;

|
clser();
n.enter1();

|
n.enter2();
n.enter3();

|
n.enter4();
n.display10();

|
n.display20();
n.display3();

|
gccch();

Input:
Enter bowler's name and no. of wickets taken:
R.Ashwin
136820

Enter batsman's name and runs made:
Virat Kohli

Enter batsman's name and ur score:
Hardik Pandya

46669

Cricketer's name:Sachin Tendulkar

Cricketer's score:136820

Bowler's name:R.Ashwin

No. of wickets taken:56

Batsman's name:Virat Kohli

No. of runs made:56000

Allrounder's name:Hardik Pandya

Allrounder's score:46669

Complete the given table:

Program Code	Write & Justify Output
<pre>a) #include <iostream.h> class Base { public: virtual void print() const = 0; }; class DerivedOne : public Base { public: void print() const { cout << "DerivedOne\n"; } }; class DerivedTwo : public Base { public: void print() const { cout << "DerivedTwo\n"; } }; class Multiple : public DerivedOne, public DerivedTwo { public: void print() const { DerivedTwo::print(); } }; int main() { int i; Multiple both; DerivedOne one; DerivedTwo two; Base *array[3]; array[0] = &both; array[1] = &one; array[2] = &two; array[1] -> print(); return 0; }</pre>	<p>In this program, 'Base' is an ambiguous base of multiple so it produces an error & this program is not virtual.</p> <p>base class</p>

References / Suggestions for further Reading

1. <https://www.geeksforgeeks.org/multiple-inheritance/>

2. <https://www.programiz.com/cpp-programming/multiple-inheritance>

Assessment Scheme

Performance Indicators	
Process related(35 Marks)	Weightage
1 Logic formulation	70%
2 Appropriate definition of multiple inheritance.	20%
3 Debugging ability	20%
4 Follow ethical practices.	10%
Product related (15 Marks)	
5 Expected Output	30%
6 Timely Submission of report	10%
7 Answer to sample questions	10%
Total (50 Marks)	100%

List of Students Team Members

1. Gajendra Singh
2. Gajendra Kumar
3. Ghunore Muskan
4. Gurav Prathmesh

Process Related(5)	Product Related(15)	Total(50)	Date
33	12	45	15/07/2023

Practical No. 14: Program to Implement Pointer to Object

I Practical Significance:

The pointer to objects helps to improve the efficiency of the program and access the objects dynamically.

II Relevant Program Outcomes (POs)

- o **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Computer engineering problem.
- o **Discipline knowledge:** Apply Computer engineering discipline - specific knowledge to solve core computer engineering related problems.
- o **Experiments and practice:** Plan to perform experiments and practices to use the results to solve broad-based Computer engineering problems.
- o **Engineering tools:** Apply relevant Computer technologies and tools with an understanding of the limitations.
- o **Communication:** Communicate effectively in oral and written form.

III Competency and Practical skills

This practical is expected to develop the following skills in you :

Develop C++ programs to solve broad-based problems

1. Define real life problems using pointer to objects.
2. Compile the program.
3. Debug and execute the program.

IV Relevant Course Outcome(s)

- Implement Inheritance in C++ program.
Use Polymorphism in C++ program.

V Practical Outcome (POs)

Write/ Compile/ debug/ Execute simple C++ program using pointer to object.

VI Relevant Affective domain related Outcome(s)

1. Select proper programming environment in C++.
2. Follow safety measures.
3. Follow ethical practices.

VII Minimum Theoretical Background

Pointers to objects:-

C++ allows you to have pointers to objects.
The pointers pointing to objects are referred to as Object Pointers.

Syntax:- class-name * object-pointer;

where class-name is the name of an already defined class and object-pointer is the pointer to an object of this class type.

For example, to declare optr as an object pointer of Sample class type, we shall write "Sample *optr", where Sample is already defined class.

When accessing members of a class using an object pointer, the arrow operator (\rightarrow) is used instead of dot operator.

III Resources required				
No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware: Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows / LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

- XII Precautions**
1. Handle computer system and peripherals with care.
 2. Follow safety practices.

X Resources used

No.	Name of Resource	Specification
1	Computer System with broad specifications	HP 280 G2, i3 Processor
2	Software	Turboc++
3	Any other resource used	

XI Result (Output of the Program)

We learnt to execute simple C++ program using pointer to object.

- XII Practical Related Questions**
Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. State output of the following code:

```
#include <iostream.h>
class myClass
{
    int i;
public:
    void read(int j)
    {
        i=j;
    }
    int getint()
    {
        return i;
    }
}
```

2. Which is the pointer which denotes the object calling the member function?

- a) Variable pointer
- b) This pointer
- c) Null pointer

3. A pointer can be initialized with _____

- a) null
- b) zero
- c) Address of an object of same type
- d) All of them

(Space for answer)

Q. Output = 10

E] This point

3.

C) Address of an object of same type

VIII Exercise:

Attempt Q1 or Q2 or Q3 and Q4 a or b from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program to declare a class "Book" containing data members book_name, author_name, and price. Accept this information for one object of the class using pointer to that object.

2. Write a C++ program to declare a class "Box" having data members height, width and breadth. Accept this information for one object using pointer to that object. Display the area and volume of that object.

3. Write a C++ program to declare a class birthday having data members day, month, year. Accept this information for five objects using pointer to the array of objects.

4. Complete the given table:

Program Code	Write & Justify Output
<pre> a) #include<iostream.h> #include<conio.h> class Time { short int hh, mm, ss; public: Time() { hh = mm = ss = 0; } void getdate(int i, int j, int k) { hh = i; mm = j; ss = k; } void prndata(void) { cout<<"\nTime is "<<hh<<"<<mm<<"<<ss<<"\n"; } }; void main() { clrscr(); Time T1, *tptr; cout<<"Initializing data members using the object, with values 12, 22, 11\n"; T1.getdata(12, 22, 11); cout<<"Printing members using the object is time using point is time "; T1.prndata(); tptr = &T1; cout<<"Printing members using the object pointer "; tptr->prndata(); cout<<"\nInitializing data members using the object pointer, with values 15, 10, 16\n"; tptr->getdata(15, 10, 16); cout<<"Printing members using the object "; T1.prndata(); cout<<"Printing members using the object pointer "; tptr->prndata(); getch(); } </pre>	<p>1. intializing data members using object with value 12, 22, 11</p> <p>2. printing members using object</p> <p>3. printing members using the object is time using point is time</p> <p>4. printing members using the object pointer</p> <p>5. printing members using the object pointer is time</p>

Name: Gaikwad Supriya Yuvraj.

Class: SY CSE.

Roll no.: 10.

/*A program to display book's information by using pointer to object*/

```

class book
{
    char b_name[10], a_name[10];
    int price;
public:
    void enter()
    {
        cout<<"Enter author's name and book's name: ";
        cin>>a_name>>b_name;
        cout<<"Enter book's price: ";
        cin>>price;
    }
    void display()
    {
        cout<<"Author's name"<<a_name;
        cout<<"\n Book's name"<<b_name;
        cout<<"\n Book's price"<<price;
    }
}

```



};

void main()

{

book b,*p;

clrscr();

p->enter();

p->display();

getch();

① gal.

Output:

Enter author's name and book's name:

Yashwant Kanetkar

Let us C

Enter book's price:

300

Author's name: Yashwant Kanetkar

Book's name: Let us C

Book's price: 300

www.niteshsolutions.xyz

```
int area , vol;
area = l*b*h;
vol = 2 * (l*b + b*h + l*b);
cout << "In area : " << area;
cout << "In volume : " << vol;
```

main()

```
b, *p;
clrscr();
� b;
� enter();
� display();
getch();
```

```

#include <iostream>
#include <string>
using namespace std;
class student
{
private:
    int rollno;
    string name;
public:
    student():rollno(0),name("") {}
    student(int r, string n) : rollno(r), name(n) {}
    void get()
    {
        cout<<"enter roll no:"; cin>>d;
        cout<<"enter name:"; cin>>n;
    }
    void print()
    {
        cout<<"roll no is " <<rollno;
        cout<<"name is "<<name;
    }
};
void putdata()
{
    cout<<"Enter birthdate is: ";
    cin>>d;
    cout<<" / " <<n;
}
void main()
{
    student *ps=new student();
    ps->get();
    ps->print();
    delete ps;
}

```

5. unable to
open file
'iosstream'

6. 'print' is
not a member
of 'student'
7. Declaration
missing.

is type-

3. class birthdate

{
int d, m, y;

public:

void getdata()

{
cout<<"Enter birthdate: ",

cin>>d>>m>>y;

3.
void putdata()

{
cout<<"Or birthdate is: " <<d <<

" / " <<n <<" / " <<y;

(Space for answers)

derived

{
3;
3;
3;
3;

birthdate B[5], *ptr;

ptr = & B[0];

for (int i=0; i<5; i++)

{

ptr->putdata();

ptr++;

{

ptr = & B[0];
for (int i=0; i<5; i++)

{

ptr->putdata();

ptr++;

{

Errors:
1) cannot
connect
'char' to xchar

2. declaration
syntax error

3. type name
expected

4. 'get' is not
member of
student

use the
specific
with an

5. unable to
open file
'iosstream'

6. 'print' is
not a member
of 'student'
7. Declaration
missing.

www.msbtresolution.xyz

```

#include <iostream>
#include <string>
using namespace std;
class student
public:
    int rollno;
    string name;
    student(int r, string n)
    {
        rollno=r;
        name=n;
    }

```

3. class birthdate

```

{
    int d, m, y;
public:
    void getdata()
    {
        cout<<"enter the birthdate:">>d>>m>>y;
    }
}

```

www.msbtresolution.xyz

```

void putdata()
{
    cout<<" birthdate is : "<<d<<
    cout<<"<<m <<y /" <<endl;
}

```

```

main()
{
    student *ps=new student();
    (*ps).get();
    (*ps).print();
    delete ps;
}

```

Errors:
 1) cannot
 convert
 (char*, to x)
 2. declaration
 syntax error

is type-
 id basic
 specific
 use the
 with an

3. type name
 expected

4. 'get' is not
 member of
 student

5. unable to
 open file
 'iosstream'

6. 'print' is
 not a member
 of student
 7. Declaration
 missing.

(Space for answers)

derived

```

{
    void major()
}

```

```

birthdate B[5], *ptr;
ptr = &B[0];
for (int i=0; i<5; i++)
{

```

```

    ptr->getdata();
}
```

```

ptr++;
}
```

```

ptr=GB[0];
for (i=0; i<5; i++)
{

```

```

    ptr->putdata();
}
```


2.

c) Class base class object as well as derived
Class object

```
res = h*w;
cout << "In Area of rectangle : " << res;
```

3.
b) Pointer

```
class triangle : public polygon
```

```
float x;
```

```
public:
```

```
void area()
```

```
{
```

```
    r=0.5*x*h;
```

```
    cout << "In Area of triangle : " << r;
```

```
}
```

```
void main()
```

```
{
```

```
    triangle();
```

```
    polygon p,*x,*ptr;
```

```
    triangle r;
```

```
    triangle t;
```

```
    ptr = &t;
```

```
    ptr->area();
```

```
((rectangle *) ptr)->area();
```

```
((triangle *) ptr)->area();
```

```
getch();
```

```
}
```

XIII Exercise

Attempt Q1 and Q3 a or b from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- Write a C++ program declare a class "polygon" having data members width and height. Derive classes "rectangle" and "triangle" from "polygon" having area() as a member function. Calculate area of triangle and rectangle using pointer to derived class object.

Maharashtra State Board of Technical Education

107

Page No.	
Date	

Page No.
Date

1. Complete the given table:

Program Code	Write & justify Output
a) #include <iostream.h> class base { public: int n1; void show() { cout<<"\n n1 = "<<n1; } }; class derive : public base { public: int n2; void show() { cout<<"\n n1 = "<<n1; cout<<"\n n2 = "<<n2; } }; int main() { base b; base *bptr; cout<<"Pointer of base class points to : " bptr=&b; bptr->n1=44; bptr->show(); derive d; cout<<"\n"; bptr=&d; bptr->n1=55; bptr->show(); return 0; }	Output : n1 = 44 n2 = 66 Points to it
b) #include <iostream.h> class BaseClass { int x; public: void setx(int i) { x = i; } int getx() { return x; } }; class DerivedClass : public BaseClass { int y; public: void sety(int i) { y = i; } };	Output : Base object x=10 Derived object y=99

~~Derived
Object~~
~~X: 99~~

```

1) Derived object
2) 99

int main()
{
    BaseClass *p; BaseClass
    baseObject; DerivedClass
    derivedObject;

    p = &baseObject;
    p->setx(10);
    cout << "Base object x: " << p->getx() <<
    '\n';
    p = &derivedObject;
    p->setx(99);

    derivedObject.sety(88);
    cout << "Derived object x: " << p->getx() <<
    '\n';
    cout << "Derived object y: " <<
    derivedObject.gety() << '\n';

    return 0;
}

```

(Space for answer)

I. Class Polygon

```

1) public:
2) int width;
3) void get();
4) cout << "Enter dimensions: ";
5) cin >> h >> w;

```

- XIV References / Suggestions for further Reading**
1. <http://www.learnprogramming.com/content/pointer-derived-class>
 2. <http://www.java2s.com/Code/Cpp/Class/Demonstratepointerderivedclass.htm>
 3. <http://www.siteforinfotech.com/2014/08/solved-mcq-on-cpp-programming-language.html>

XV Assessment Scheme

Performance indicators		Weightage
Process related (35 Marks)		70%
1 Logic formation		20%
2 Appropriate use of pointer to derived class		20%
3 Debugging ability		20%
4 Follow ethical practices.		10%
Product related (15 Marks)		30%
5 Expected Output		10%
6 Timely Submission of report		10%
7 Answer to sample questions		10%
	TOTAL	100%

List of Students Team Members

1. Gaurav Khandekar
2. Gopal Komal
3. Ghunake Mukund
4. Gurav Prathmesh.

Marks Obtained	Dated signature of Teacher
Process Related(35) 33	Product Related(15) 12

**Total Marks
Obtained**

45

Object Oriented Programming Using C++ (22316)

```

public:// required constructor
Distance( float feet, float inches ) {
    Distance( int feet, int inches ) {
        feet = feet;
        inches = inches;
    }
    void displayDistance() {
        cout << "feet " << feet << " inches " << inches << endl;
    }
    // overloading minus (-) operator
    Distance operator-( float feet ) {
        inches -= inches;
        return Distance(feet, inches);
    }
}
int main()
{
    Distance d1(1.0f, 0.2f);
    d1.displayDistance();
}

```

VIII. Required		IX. Precautions		X. Object Oriented Programming Using C++	
Sr. No.	Name of Resource	Specification	Quantity	Remarks	
1	Computer System with broad specifications	Computer (i3-15 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments	
2	Operating system	Windows /LINUX			
3	Software	Turbo C++ Version 3.0 or any other			

Resources used		Specification			
S. No.	Name of Resource	Specification			
1	Computer System	HP 280 G2 i3 Processor			
2	Software	Turbo C++			
3	Any other resource used	—			

Object Oriented Programming

No.	Resource	Computer (US\$)	RAM preferable	Processor minimum 1 GB and converts, iPD, 40GHz and above	As per batch	For all size	Experiments
1	Hardware: Computer System						
2	Operating system	Windows (LINUX)					
3	Software	Turbo C++ Version 3.0 or any other					
Precautions							Precautions Handle computer system and peripherals with care. Follow safety practices.
Resources used							
S. No.	Name of Resource	Specification					
1	Computer System with broad specifications	hp	2.80	4T2. 1/3	Processor		
2	Software		TURBO C++				
3	Any other resource used		—				

XII Practical Related Questions

Practical Related Questions *Practical Related Questions* are given for reference. Teacher must design *Note: Below given are few sample questions for achievement of identified C.O.*

*Note: Answer
more such questions so as to ensure ins-
tance X and XIII to XV for all relevant progra-
(Note: Use Point VII to X and XIII to XV.)*

Note: Use Point VIII to X and XIII to XV for all relevant programming. Answer more such questions so as to ensure in-
(Note: Use Point VIII to X and XIII to XV for all relevant programming. Answer more such questions so as to ensure in-

(1) What is function overloading and operator overloading?

(1) What is function overloading and operator overloading? State output of the following code:

class Distance (

class Distance {

Maharashtra State Board of Technical Education

2. Complete the given table:

Program Code	Write & Justify Output
a)	error:- 1.'s' Expected 2. statement missing:
b)	3. undefined deleined symbol 'pt1' 4. undefined symbol 'p+2'
c)	Output E:-11 I:-10 F:-5 I:-11

XIII Exercise

Attempt Q1 and Q3 a or b from the following:

- (Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)
- Write a C++ program to overload unary operators (++) increment and (--)decrement.

#include <iostream.h>

{

private:

double m_x, m_y, m_z;

public:

Point(double x=0.0, double y=0.0, double z=0.0):

m_x(x), m_y(y), m_z(z)

{}

Point operator- () const;

bool operator! () const;

double getX() { return m_x; }

double getY() { return m_y; }

double getZ() { return m_z; }

};

Point Point::operator- () const

{

return Point(-m_x, -m_y, -m_z);

}

bool Point::operator! () const

{

return (m_x == 0.0 && m_y == 0.0 && m_z == 0.0);

}

int main()

{

Point point; // use default constructor to set

to (0.0, 0.0, 0.0)

if (!point)

std::cout << "point is set at the

origin.\n";

else

std::cout << "point is not set at the

origin.\n";

return 0;

}

(Space for answers)

Errors:
J-type
Name
except

Ques.

Name: Gaikwad supriya yuvraj
Roll no: 10
class include
int x, y;
public:
Void enter()
Void enter0()

Cout<<" enter two numbers ;
Cin>x>y;
X-;
Y-;
Void operator ++()

Void operator --()

Void operator +()

Void operator -()

Void operator *()

Void operator /()

Void operator %()

Void operator ==()

Void operator !=()

Void operator <=()

Void operator >=()

Void operator <()

Void operator >()

Void operator ~()

Void operator !()

Void operator &()

Void operator &&()

Void operator ||()

Void operator =()

Void operator +=()

Void operator -=()

Void operator *=()

Void operator /=()

Void operator %=()

Void operator <<()

Void operator >>()

Void operator <<<()

Void operator >>>()

Void operator <<<<()

Void operator >>>>()

Void operator <<<<<()

Void operator >>>>>()

Void operator <<<<<<()

Void operator >>>>>>()

Void operator <<<<<<<()

Void operator >>>>>>>()

Void operator <<<<<<<<()

Void operator >>>>>>>>()

Void operator <<<<<<<<<()

Void operator >>>>>>>>>()

Void operator <<<<<<<<<<()

Void operator >>>>>>>>>>()

#include <iostream.h>
class Date
{
public:
 int Day; // day of month
 int Month; // month (1=Jan, 12=Dec)
 int Year; // year
};

cout << "After decrement object of type " ;
Date D;

D--;

cout << "After increment object twice " ;
Date D;
D++;
D++;

① ~~Display~~

www.msbtresolution.xyz

J. cl.
3

Old Specie in C

244

944

old display()

```
cout << " << cout << q << endl;
```

```
{ main()
    Inode n;
    dirent r;
    ~n();
    subroot
        cout << "In A file directory object of type",
        n.display();
    ~r();
}
```

```
cout << "In FILE implement object type",
n.display();
getch();
```

3

www.msbtesolution.xyz

References / Suggestions for further Reading

- http://www.silverbook.com/~vivek/compfund/using_stdio_operations.htm
- http://www.karunya.edu/~prakash/03-04/lectures/03-04-02-stdio.htm
- http://www.tutorialspoint.com/c_stdio.htm
- http://www.tutorialspoint.com/c_stdio.htm
- http://www.vishwakarma.com/Computer%20Engineering/programme.htm
- http://www.vishwakarma.com/Computer%20Engineering/programme.htm
- http://www.vishwakarma.com/Computer%20Engineering/programme.htm

Step 4: Using the function operator +() to add two complex numbers.

Step 5: Define the function operator -() to subtract two complex numbers.

Step 6: Define the function display function.

Step 7: Define the class objects obj1,obj2 and numbers.

Step 8: Declare the class objects get value using obj1 and obj2.

Step 9: Call the function get value by calling the result.

Step 10: Calculate the value for the object result by calling the function operator + and operator -.

Step 11: Call the display function using obj1 and obj2 and result.

Step 12: Return the values.

Step 13: Stop the program.

VIII Resources required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware: Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows/LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- Handle computer system and peripherals with care.
- Follow safety practices.

X Resources used

S.No.	Name of Resource	Specification
1	Computer System with broad specifications	hp 280 G2 , i5 processor
2	Software	Turbo C++
3	Any other resource used	?

XI Result (Output of the Program)

Code learnt to execute simple C++ program using binary operator overloading.

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. What is the difference between unary operator overloading and binary operator overloading in C++?

2. State output of the following code:

```
#include<iostream.h>
class overloading {
    int value;
public:
    void setValue(int temp) {
        value = temp;
    }
    overloading operator+(overloading ob) {
        t.value = value + ob.value;
    }
    void display() {
        cout << value << endl;
    }
}
```

```
void main() {
    overloading obj1, obj2, result;
    int a, b;
    cout << "Enter the value of Complex Numbers a,b:" ;
    obj1.setValue(a);
    obj2.setValue(b);
    result = obj1 + obj2;
    cout << "Input Values:\n";
    obj1.display();
    obj2.display();
    cout << "Result:" ;
    result.display();
    getch();
}
```

```

int n = num*f.deno; int d = demo*f.num; return
Fraction(n/gcd(n,d),d/gcd(n,d));
}
bool operator == (Fraction &f)
{
    if(n == f.num && d == f.deno) int d = demo*f.num; return
Fraction(n/gcd(n,d),d/gcd(n,d));
}
bool operator != (Fraction &f)
{
    if(n == f.num && d == f.deno);
}
int gcd(int n, int d)
{
    int rem;
    while (d != 0)
    {
        rem = n % d;
        n = d;
        d = rem;
    }
    return n;
}
void accept()
{
cout<<"\n Enter Numerator : ";
cin>>num;
cout<<"\n Enter Denominator : ";
cin>>deno;
}
void main()
{
Fraction f1; Fraction f2; Fraction f3;
cout<<"\n Enter 1st Fraction Value "; cout<<"\n ----
-----\n"; f1.accept();
cout<<"\n Enter 2nd Fraction Value "; cout<<"\n ----
-----\n"; f2.accept();
f3=f1+f2;
cout<<"\n Sum of Two Numbers
-----\n";
cout<<f3.num<<" / "<<f3.deno<<endl;
}

```

10. Illegal Structure Operation

f.deno is not member of C math.

12. Illegal Structure Operation.

```

if(f1 == f2)
cout<<"\n Fraction 1 is Equal to
Fraction 2"\>>endl;
else
cout<<"\n Fraction 1 is Not Equal to
Fraction 2"\>>endl;
}

```

1. Type name expected
Errors:

2. Declaration missing;

3. Illegal Structure Declaration.

```

b) #include <iostream.h>
class Box {
double length;
double breadth;
double height;
public:
    double getVolume(void) {
        return length * breadth * height;
    }
    void setLength( double len ) {
        length = len;
    }
    void setHeight( double hei ) {
        height = hei;
    }
    void setBreadth( double bre ) {
        breadth = bre;
    }
};

Box operator+(const Box& b) {
    Box box;
    box.length = this->length + b.length;
}
```

objects.

Box operator+(const Box& b) {

box.box;

box.length = this->length + b.length;

```

int n = r
Fraction
}
bool operator<(Box a, Box b)
{
    if (a.height < b.height)
        return true;
    else
        return false;
}

int main()
{
    Box Box1, Box Box2, Box Box3;
    double volume = 0.0;
    Box1.setLength(6.0); Box1.setBreadth(7.0);
    Box1.setHeight(5.0);
    Box2.setLength(12.0); Box2.setBreadth(13.0);
    Box2.setHeight(10.0);
    Box2.setBreadth(10.0);

    volume = Box1.getVolume();
    cout << "Volume of Box1 : " << volume << endl;

    volume = Box2.getVolume();
    cout << "Volume of Box2 : " << volume << endl;

    Box3 = Box1 + Box2;
    cout << "Volume of Box3 : " << volume << endl;
    cout << "Volume of Box3 : " << volume << endl;
    return 0;
}

```

(Space for answers)

2.

```

class binary
{
    float a;
    int b;
public:
    void set (float x)
    {
        a=x;
    }
    void show()
    {
        cout << "In a = " << a << endl;
        cout << "In b = " << b << endl;
    }
};

binary operator + (binary b2)
{
    binary temp;
    temp.a = b2.a + a;
    temp.b = b2.b + b;
    return temp;
}

binary operator - (binary b2)
{
    binary temp;
    temp.a = b2.a - a;
    temp.b = b2.b - b;
    return temp;
}

```

binary

operator * (binary.b2)

```
binary temp;
temp.a = b2.a * a;
return temp;
```

binary operator / (binary b2)

```
binary temp;
temp.a = b2.a/a;
return temp;
```

void main()

```
binary b1, b2, b3;
clrscr();
```

```
b1.set(10,5);
b2.set(20,5);
b3.show();
```

```
b3.show();
```

```
cout << "in operator + overloading:";
```

```
b3 = b1.operator + b2;
cout << "in operator - overloading:";
```

```
b3 = b1.operator - b2;
b3.show();
```

(out << "in operator * overloading:";
b3 = b1.operator * b2;

b3.show());

(out << "in operator / overloading:";
b3 = b1.operator / b2;

b3.show();
qexit();

class compare {
char str[20];
int str1;

public:
void get() {

cout << "in enter string:";

cin>>str;

3 void show()

{ cout << "in entered string is:";

3 << str;

compare operator== compare,

compare operator<= compare,

compare operator>= compare,

compare operator<> compare,

binary operators * (bitwise)

binary temp
 $\text{temp} \cdot a = b_2 \cdot a * a;$
 return temp;
 3
 binary operator / ()

```
cout << "In operator * overloading";
b3 = b1 . operator * . b2;
b3 . show();
cout << "In operator / overloading";
b3 = b1 . operator / . b2;
b3 . show();
qerch();
```

3. **Q. main()**

```
binary temp;
temp.a = b2.a/b1;
return temp;
```

Q. main()

```
b1.name b1, b2, b3;
cout << b1;
b1.set(10,5);
b2.set(20,5);
b3.show();
cout << endl;
cout << "In entered string:";
cin >> str;
3
void show()
{
    cout << " In entered string is:";
    << str;
}
compare operator== compare
{
    compare temp;
    temp.str1 = strcmp(str1, c2.str1);
}
```

class compare.

char str[20];

int str1;

public:

void get()

{

cout << " In entered string:";

cin >> str;

}

void show()

{

cout << " In entered string is:";

<< str;

}

compare operator== compare

{

compare temp;

temp.str1 = strcmp(str1, c2.str1);

add operator +(add a1, add a2)

add temp;

temp.a=a2.a+a1.a;

temp.b=a2.b+a1.b;

return temp;

}

void main()

{

add a1,a2,a3;

clrscr();

a1.setdata(14.5,2.5);

a2.setdata(35.5,45.5);

a3=operator +(a2,a1);

a3.show();

getch();

}

Output:

⑤ *Output*

a:50

b:48

onal Overloading

one definitions for a concept of polymorphism.

aties, sciences and basic engineering problem. A discipline - specific problems, and practices to use the graphics and tools with an often form.

u:

same scope to implement

tion overloading.

e functions can have

e number, types and commonly used to similar tasks but on morphism feature in

- www.tutorialspoint.com/cpp/operator_overloading_by_using_friend_function.htm
- www.tutorialspoint.com/cpp/operator_overloading_programs/overloadings_demonstrating_operator_overloading_by_using_friend_function.htm
- www.tutorialspoint.com/cpp/operator_overloading_programs/comparing_and_extraction_and_extraction_operators.htm
- www.tutorialspoint.com/cpp/operator_overloading_programs/comparing_arithmetic_operations_on_two_fractions.htm
- www.includeshelp.com/cpp_programs_to_add_two_numbers_using_binary_plus_operator_overloading.aspx

(Showing for answers)

4. Complete the given table:

Program Code	Write & justify Output
<pre>a) #include<iostream.h> int main() { float absvalue(); int main() { int a = -5; float b; cout << "Value of a = " << a << endl; cout << "Value of b = " << b << endl; cout << "Absolute value of a = " << absvalue(a); cout << "Absolute value of b = " << absvalue(b); return 0; } float absvalue(float var) { if (var < 0) var = -var; return var; } }</pre>	<p>Output:- Absolute value of -5 = 5</p> <p>Absolute value of 5.5 = 5.5</p>

XIII Exercise

Attempt Q1 or Q2 or Q3 and Q4 a or b from the following:

(Note: Use Page VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ Program to interchange the values of two int , float and char

using function overloading

points in 2D and 3D

www.msbtessolution.xyz

Syntax

```
class class_Name
{
    returntype method()
    .....
}
returntype method(datatype1 variable1)
{
    .....
}
```

```
returntype method(datatype1 variable1, datatype2 variable2)
{
    .....
};
```

Examples

```
void display(); //function with no arguments
void display( int ); //function with one integer type arguments
void display( float ); //function with one floating point arguments
void display( int, float ); //function with one floating and one integer type argument
```

VIII Resources required

Sr. No.	Name of Resource	Specification	Quantity	Remarks
1	Hardware Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards, HDD 40GB and above	As per batch size	For all Experiments
2	Operating system	Windows / LINUX		
3	Software	Turbo C++ Version 3.0 or any other		

IX Precautions

- 1 Handle computer system and peripherals with care.
- 2 Follow safety practices.

X Resources used

S. No.	Name of Resource	Specification
1	Computer System with broad specifications	HP 280 G2, i3 processor
2	Software	Turbo C++
3	Any other resource used	

XI Result (Output of the Program)

Using input to execute simple C++ program

XII Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

- (Note: Use Point VII to X and XII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)
1. Does constructor overloading is implementation of function overloading?
 2. State output of the following code:
- ```
#include <iostream.h>

int operate (int a, int b)
{
 return (a * b);
}

float operate (float a, float b)
{
 return (a / b);
}

int main()
{
 int x = 5, y = 2;
 float n = 5.0, m = 2.0;
 cout << operate(x, y) << endl;
 cout << operate (n, m);
 return 0;
}
```

3. Which of the following in Object Oriented Programming is supported by Function overloading and default arguments features of C++?
  - a) Inheritance
  - b) Polymorphism
  - c) Encapsulation
  - d) None of these
4. Overloaded functions are
  - a) Very long functions that can hardly run
  - b) One function containing another one or more functions inside it.
  - c) Two or more functions with the same name but different number parameters
  - d) None of these

(Space for answers)

4. Complete the given table:

| Program Code                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Write & justify<br>Output                                                                              |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| <pre>a) #include&lt;iostream.h&gt; int absolute(int); float absolute(float);  int main() {     int a = -5;     float b = 5.5;     cout &lt;&lt; "Absolute value of " &lt;&lt; a &lt;&lt; " = " &lt;&lt;     absolute(a) &lt;&lt; endl;     cout &lt;&lt; "Absolute value of " &lt;&lt; b &lt;&lt; " = " &lt;&lt;     absolute(b);     return 0; }  int absolute(int var) {     if (var &lt; 0)         var = -var;     return var; }  float absolute(float var) {     if (var &lt; 0.0)         var = -var;     return var; }</pre> | <p>Absolute Value of<br/> <math>-5 = 5</math></p> <p>Absolute Value of<br/> <math>5.5 = 5.5</math></p> |

**XIII Exercise**

Attempt Q1 or Q2 or Q3 and Q.4 a or b from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ Program to interchange the values of two int , float and char using function overloading .
2. Write a C++ Program that find the distance between two points in 2D and 3D space using function overloading.
3. Write C++ program to find the area of various geometrical shapes by function overloading.

三

### class distance

int 3- 3-13-3- V2-4-4-143-32-21-23;

```
public:
```

~

$$d = 891 + ((c_2, -x_0) * (x_1, -x_0)) + (41 - 40) *$$

counts" in Benders two-point **optimization**.

卷之三

```
void dist (int x2, int y2, int z2)
```

float di

$$d = \sqrt{(x_2 - x_3)^2 + (y_2 - y_3)^2}$$

points in 3D space is: "ccdi".

卷之三

```
void main()
```

1

distance obj

卷之三

obj. dist (2,4,6,8);

Maharashtra

Object

Output : Distance between two points in 2D space  
Is : 1.732050  
Distance between two Points in 2D space  
Is: 2.828427.

3. class geometric

```
public :
 int A, b, r, area;
 float pi, h, ba;
```

public :

```
void get (int x, int y)
{
 A = x;
 b = y;
}
```

```
length() " << " breadth() " << endl;
cout << " Area = " << area;
cout << " Circumference = " << circumference();
cout << " In radius = " << r;
```

```
void print (float p, int r)
{
 pi = p;
 r = 4;
 cout << " In radius = " << r;
 area = pi * r * r;
 cout << " In area of circle = " << area;
}
```

```
void main()
```

www.msbtatesolution.kyz

/\*To swap values of 2 int,float,char using function overloading\*/

class swap

{

public:

int a,b,temp1;

float c,d,temp2;

char e,f,temp 3;

public:

void get(int x,int y)

{

a=x;

b=y;

cout<<"\n Values before int swapping:";

cout<<"\n a:<<a<<" b:<<b;

temp1=a;

a=b;

b=temp1;

cout<<"\n Values after int swapping are:";

cout<<"\na."<<a<<" b:<<b;

```

Object
;

void set(float p,float q)
{
 c=p;
 d=q;
}

cout<<"\n Values before float swapping:";

cout<<c:<<d<<"\n Values after float swapping:";

swap n;
main()
{
 clrscr();
 n.get(10.15);
 n.set(10.5,15.5);
 n.wet('s','g');
 getch();
}
}

void we(char u,char v)
{
 e=u;
 f=v;
}

cout<<"\n Values before char swapping:";

cout<<c:<<e<<"\n Values after char swapping are:";

e=f;
f=temp3;
}

```

www.msbtresolution.xyz

Output:

Values before float swapping:  
 cout<<c:<<d<<"  
 10.5 15.5  
 Values after float swapping:  
 cout<<c:<<e<<"  
 15.5 10.5  
 Values after char swapping are:  
 cout<<c:<<e<<"  
 c:15.5 b:10.5  
 Values after char swapping are:  
 cout<<c:<<e<<"  
 c:10.5 b:15.5  
 Values after float swapping:  
 cout<<c:<<d<<"  
 c:15.5 b:10.5  
 Values after float swapping:  
 cout<<c:<<d<<"  
 c:10.5 b:15.5

卷之三

卷之三

四  
卷之二

5

Digitized by srujanika@gmail.com



| Marks Obtained           |                        |           | Dated signature<br>of Teacher |
|--------------------------|------------------------|-----------|-------------------------------|
| Practical<br>Related(55) | Product<br>Related(55) | Total(50) |                               |
| 33                       | 12                     | 45        | Opal                          |

- IX Precautions**
- Handle computer system and peripherals with care.
  - Follow safety practices.

**X Resources used**

| S. No. | Name of Resource                          | Specification            |
|--------|-------------------------------------------|--------------------------|
| 1      | Computer System with broad specifications | hp 280 G2 , i3 processor |
| 2      | Software                                  | Turbo C++ ✓              |
| 3      | Any other resource used                   | ✓                        |

**XI Result (Output of the Program)**

We learn to write a program to read data from file.

**XII Practical Related Questions**

*Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.*

(Note: Use Point VIII to X and XII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

- What are file handling classes in C++?
- State output of the following code:

```
#include<iostream.h>
#include<iomanip.h>
#include<string.h>
#include<cctype.h>

int main()

{
 ifstream ifile;
 ifile.open ("text.txt");
 cout << "Reading data from a file :-" << endl ;
 int c = ifile.peek();
 if (c == EOF) return 1;
 if (!isdigit(c))
 {
 cout << "Data in the file: " << endl;
 cout << endl;
 }
 else
 {
 string str;
 ifile >> str;
 cout << "Data in the file: " << str << endl;
 }
}
```

Name:Supriya Yuvraj Gaikwad.

Class:SY CSE.

Roll no.:10

/\*To merge two files\*/

```
#include<iostream.h>
void main()
{
 ifstream ifiles1,ifiles2;
 ofstream ifiler;
 clrscr();
 ifiles1.open("file1");
 ifiles2.open("file2");
 ifile.open("file3");
 ifile >> name1 >> name2 >> name3;
 cout << endl;
 cout << "Enter first file name:";
```

```
gets(name1);
cout << endl;
cout << "Enter second file name:";
```

```
gets(name2);
cout << endl;
cout << "Enter name of file which will store the contents of above
two files:";
```

```
gets(name3);
ifiles1.open(name1);
ifiles2.open(name2);
```

```

 {
 perror("Error message");
 cout<<"\nPress any key to exit... ";
 getch();
 exit(EXIT_FAILURE);
 }
 while(files1.eof()==0)
 {
 X1 Re
 {
 ifiles1>>ch;
 ifilet<<ch;
 }
 while(ifiles2.eof()==0)
 {
 files2>>ch;
 ifilet<<ch;
 }
 cout<<"\nThe two files are merged!";
 ifiles1.close();
 ifiles2.close();
 ifilet.close();
 getch();
 }
}

```

Output:

Enter first file name:myfile1  
Enter second file name:myfile2

Enter name of file which will store the contents of above two files:myfile3

The two files are merged!

[www.msbtresolution.xyz](http://www.msbtresolution.xyz)

```
ifile.close();
}
return 0;
```

3. State output of the following code:

```
#include<iostream.h>
#include<fstream.h>
#include<cctype.h>
int main ()
{
 ifstream ifile;
 ifile.open ("text.txt");
 char last;
 ifile.ignore (256, ' ');
 cout << "Your initial is " << last << '\n';
 ifile.close();
 return 0;
}
```

4. Which operator is used to insert the data into file?

- a) >>
- b) <<
- c) <
- d) none of the mentioned

5. Which function is used to position back from the end of file object?

- a) seekg()
- b) seekp()
- c) both seekg() & seekp()
- d) none of the mentioned

-TJQ4LJO  
(Space for answers)

### Some Standard file handling classes

1) fstream

2) ifstream

3) ofstream

4) (b) <<

5) >>

6) (c) both seekg() & seekp()

7) Errors -  
function 'isdigit' should have a prototype.

2. Misplaced else

**XII** Exercise

Attempt any 2 from (1-4) and Q5-a & b from the following:

(Note: Use Point VIII to X and XIII to XV for all relevant programming exercise use blank pages provided or attach more pages if needed.)

1. Write a C++ program ask to the user to enter file name to encrypt its content.
2. Write a C++ program to merge two files.
3. Write a C++ program to Read and Display File's Content.
4. Write a C++ program to List Files in Current Directory.
5. Complete the given table:

| Program Code                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Write & justify                                                                                                                                                                                                                                        |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Output                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                                                                                                                                                                                                                                        |
| a) #include<iostream.h><br>int main()<br>{<br>fstream fpt; // file pointer<br>fstream file;<br>file.open("text.txt");<br>file << "geeksforgeeks" << endl;<br>file << "Data written to file" << endl;<br>file.close();<br>return 0;<br>}<br><br>b) #include <iostream.h><br>#include<iostream.h><br>int main()<br>{<br>char data[100];<br>ifstream file;<br>//create a text file before executing<br>//file.open ("text.txt");<br>while (!file.eof())<br>{<br>file.getline(data, 100);<br>if(file.eof())<br>break;<br>cout << data << endl; // outputs<br>// "geeksforgeeks"<br>// "Data written to file"<br>}<br>} | Output -<br>Data written<br>to file +<br>also prints<br>'geeks for geeks'<br>in file test.txt<br><br>cout << "In Error in opening file!"<br>getline();<br>getch();<br>cout << "Press key to exit!";<br>cout << endl;<br>cout << endl;<br>cout << endl; |

```
10 encrypt file * /
11 include <iostream.h>
12 #include <fstream.h>
13 void main()
14 {
15 clrscr();
16 char fname [20], ch, choice;
17 fstream Fps, Fpt;
18 cout << "In encrypt file name:";
19 gets (fname);
20 Fps = open (fname);
21 if (!Fps) {
22 cout << "Error in opening file!"
23 exit (1);
24 }
25 cout << "File opened successfully";
26 cout << endl;
27 cout << "Enter choice 1 for encryption and 2 for decryption";
28 cin >> choice;
29 if (choice == 1)
30 {
31 cout << "Enter file name for encryption";
32 cin >> fname;
33 cout << endl;
34 cout << "Enter file name for decryption";
35 cin >> fname2;
36 cout << endl;
37 cout << "Enter file name for output";
38 cin >> fname3;
39 cout << endl;
40 cout << "Enter file name for error";
41 cin >> fname4;
42 cout << endl;
43 cout << "Enter file name for key";
44 cin >> fname5;
45 cout << endl;
46 cout << "Enter file name for key2";
47 cin >> fname6;
48 cout << endl;
49 cout << "Enter file name for key3";
50 cin >> fname7;
51 cout << endl;
52 cout << "Enter file name for key4";
53 cin >> fname8;
54 cout << endl;
55 cout << "Enter file name for key5";
56 cin >> fname9;
57 cout << endl;
58 cout << "Enter file name for key6";
59 cin >> fname10;
60 cout << endl;
61 cout << "Enter file name for key7";
62 cin >> fname11;
63 cout << endl;
64 cout << "Enter file name for key8";
65 cin >> fname12;
66 cout << endl;
67 cout << "Enter file name for key9";
68 cin >> fname13;
69 cout << endl;
70 cout << "Enter file name for key10";
71 cin >> fname14;
72 cout << endl;
73 cout << "Enter file name for key11";
74 cin >> fname15;
75 cout << endl;
76 cout << "Enter file name for key12";
77 cin >> fname16;
78 cout << endl;
79 cout << "Enter file name for key13";
80 cin >> fname17;
81 cout << endl;
82 cout << "Enter file name for key14";
83 cin >> fname18;
84 cout << endl;
85 cout << "Enter file name for key15";
86 cin >> fname19;
87 cout << endl;
88 cout << "Enter file name for key16";
89 cin >> fname20;
90 cout << endl;
91 cout << "Enter file name for key17";
92 cin >> fname21;
93 cout << endl;
94 cout << "Enter file name for key18";
95 cin >> fname22;
96 cout << endl;
97 cout << "Enter file name for key19";
98 cin >> fname23;
99 cout << endl;
100 cout << "Enter file name for key20";
101 cin >> fname24;
102 cout << endl;
103 cout << "Enter file name for key21";
104 cin >> fname25;
105 cout << endl;
106 cout << "Enter file name for key22";
107 cin >> fname26;
108 cout << endl;
109 cout << "Enter file name for key23";
110 cin >> fname27;
111 cout << endl;
112 cout << "Enter file name for key24";
113 cin >> fname28;
114 cout << endl;
115 cout << "Enter file name for key25";
116 cin >> fname29;
117 cout << endl;
118 cout << "Enter file name for key26";
119 cin >> fname30;
120 cout << endl;
121 cout << "Enter file name for key27";
122 cin >> fname31;
123 cout << endl;
124 cout << "Enter file name for key28";
125 cin >> fname32;
126 cout << endl;
127 cout << "Enter file name for key29";
128 cin >> fname33;
129 cout << endl;
130 cout << "Enter file name for key30";
131 cin >> fname34;
132 cout << endl;
133 cout << "Enter file name for key31";
134 cin >> fname35;
135 cout << endl;
136 cout << "Enter file name for key32";
137 cin >> fname36;
138 cout << endl;
139 cout << "Enter file name for key33";
140 cin >> fname37;
141 cout << endl;
142 cout << "Enter file name for key34";
143 cin >> fname38;
144 cout << endl;
145 cout << "Enter file name for key35";
146 cin >> fname39;
147 cout << endl;
148 cout << "Enter file name for key36";
149 cin >> fname40;
150 cout << endl;
151 cout << "Enter file name for key37";
152 cin >> fname41;
153 cout << endl;
154 cout << "Enter file name for key38";
155 cin >> fname42;
156 cout << endl;
157 cout << "Enter file name for key39";
158 cin >> fname43;
159 cout << endl;
160 cout << "Enter file name for key40";
161 cin >> fname44;
162 cout << endl;
163 cout << "Enter file name for key41";
164 cin >> fname45;
165 cout << endl;
166 cout << "Enter file name for key42";
167 cin >> fname46;
168 cout << endl;
169 cout << "Enter file name for key43";
170 cin >> fname47;
171 cout << endl;
172 cout << "Enter file name for key44";
173 cin >> fname48;
174 cout << endl;
175 cout << "Enter file name for key45";
176 cin >> fname49;
177 cout << endl;
178 cout << "Enter file name for key46";
179 cin >> fname50;
180 cout << endl;
181 cout << "Enter file name for key47";
182 cin >> fname51;
183 cout << endl;
184 cout << "Enter file name for key48";
185 cin >> fname52;
186 cout << endl;
187 cout << "Enter file name for key49";
188 cin >> fname53;
189 cout << endl;
190 cout << "Enter file name for key50";
191 cin >> fname54;
192 cout << endl;
193 cout << "Enter file name for key51";
194 cin >> fname55;
195 cout << endl;
196 cout << "Enter file name for key52";
197 cin >> fname56;
198 cout << endl;
199 cout << "Enter file name for key53";
200 cin >> fname57;
201 cout << endl;
202 cout << "Enter file name for key54";
203 cin >> fname58;
204 cout << endl;
205 cout << "Enter file name for key55";
206 cin >> fname59;
207 cout << endl;
208 cout << "Enter file name for key56";
209 cin >> fname60;
210 cout << endl;
211 cout << "Enter file name for key57";
212 cin >> fname61;
213 cout << endl;
214 cout << "Enter file name for key58";
215 cin >> fname62;
216 cout << endl;
217 cout << "Enter file name for key59";
218 cin >> fname63;
219 cout << endl;
220 cout << "Enter file name for key60";
221 cin >> fname64;
222 cout << endl;
223 cout << "Enter file name for key61";
224 cin >> fname65;
225 cout << endl;
226 cout << "Enter file name for key62";
227 cin >> fname66;
228 cout << endl;
229 cout << "Enter file name for key63";
230 cin >> fname67;
231 cout << endl;
232 cout << "Enter file name for key64";
233 cin >> fname68;
234 cout << endl;
235 cout << "Enter file name for key65";
236 cin >> fname69;
237 cout << endl;
238 cout << "Enter file name for key66";
239 cin >> fname70;
240 cout << endl;
241 cout << "Enter file name for key67";
242 cin >> fname71;
243 cout << endl;
244 cout << "Enter file name for key68";
245 cin >> fname72;
246 cout << endl;
247 cout << "Enter file name for key69";
248 cin >> fname73;
249 cout << endl;
250 cout << "Enter file name for key70";
251 cin >> fname74;
252 cout << endl;
253 cout << "Enter file name for key71";
254 cin >> fname75;
255 cout << endl;
256 cout << "Enter file name for key72";
257 cin >> fname76;
258 cout << endl;
259 cout << "Enter file name for key73";
260 cin >> fname77;
261 cout << endl;
262 cout << "Enter file name for key74";
263 cin >> fname78;
264 cout << endl;
265 cout << "Enter file name for key75";
266 cin >> fname79;
267 cout << endl;
268 cout << "Enter file name for key76";
269 cin >> fname80;
270 cout << endl;
271 cout << "Enter file name for key77";
272 cin >> fname81;
273 cout << endl;
274 cout << "Enter file name for key78";
275 cin >> fname82;
276 cout << endl;
277 cout << "Enter file name for key79";
278 cin >> fname83;
279 cout << endl;
280 cout << "Enter file name for key80";
281 cin >> fname84;
282 cout << endl;
283 cout << "Enter file name for key81";
284 cin >> fname85;
285 cout << endl;
286 cout << "Enter file name for key82";
287 cin >> fname86;
288 cout << endl;
289 cout << "Enter file name for key83";
290 cin >> fname87;
291 cout << endl;
292 cout << "Enter file name for key84";
293 cin >> fname88;
294 cout << endl;
295 cout << "Enter file name for key85";
296 cin >> fname89;
297 cout << endl;
298 cout << "Enter file name for key86";
299 cin >> fname90;
300 cout << endl;
301 cout << "Enter file name for key87";
302 cin >> fname91;
303 cout << endl;
304 cout << "Enter file name for key88";
305 cin >> fname92;
306 cout << endl;
307 cout << "Enter file name for key89";
308 cin >> fname93;
309 cout << endl;
310 cout << "Enter file name for key90";
311 cin >> fname94;
312 cout << endl;
313 cout << "Enter file name for key91";
314 cin >> fname95;
315 cout << endl;
316 cout << "Enter file name for key92";
317 cin >> fname96;
318 cout << endl;
319 cout << "Enter file name for key93";
320 cin >> fname97;
321 cout << endl;
322 cout << "Enter file name for key94";
323 cin >> fname98;
324 cout << endl;
325 cout << "Enter file name for key95";
326 cin >> fname99;
327 cout << endl;
328 cout << "Enter file name for key96";
329 cin >> fname100;
330 cout << endl;
331 cout << "Enter file name for key97";
332 cin >> fname101;
333 cout << endl;
334 cout << "Enter file name for key98";
335 cin >> fname102;
336 cout << endl;
337 cout << "Enter file name for key99";
338 cin >> fname103;
339 cout << endl;
340 cout << "Enter file name for key100";
341 cin >> fname104;
342 cout << endl;
343 cout << "Enter file name for key101";
344 cin >> fname105;
345 cout << endl;
346 cout << "Enter file name for key102";
347 cin >> fname106;
348 cout << endl;
349 cout << "Enter file name for key103";
350 cin >> fname107;
351 cout << endl;
352 cout << "Enter file name for key104";
353 cin >> fname108;
354 cout << endl;
355 cout << "Enter file name for key105";
356 cin >> fname109;
357 cout << endl;
358 cout << "Enter file name for key106";
359 cin >> fname110;
360 cout << endl;
361 cout << "Enter file name for key107";
362 cin >> fname111;
363 cout << endl;
364 cout << "Enter file name for key108";
365 cin >> fname112;
366 cout << endl;
367 cout << "Enter file name for key109";
368 cin >> fname113;
369 cout << endl;
370 cout << "Enter file name for key110";
371 cin >> fname114;
372 cout << endl;
373 cout << "Enter file name for key111";
374 cin >> fname115;
375 cout << endl;
376 cout << "Enter file name for key112";
377 cin >> fname116;
378 cout << endl;
379 cout << "Enter file name for key113";
380 cin >> fname117;
381 cout << endl;
382 cout << "Enter file name for key114";
383 cin >> fname118;
384 cout << endl;
385 cout << "Enter file name for key115";
386 cin >> fname119;
387 cout << endl;
388 cout << "Enter file name for key116";
389 cin >> fname120;
390 cout << endl;
391 cout << "Enter file name for key117";
392 cin >> fname121;
393 cout << endl;
394 cout << "Enter file name for key118";
395 cin >> fname122;
396 cout << endl;
397 cout << "Enter file name for key119";
398 cin >> fname123;
399 cout << endl;
400 cout << "Enter file name for key120";
401 cin >> fname124;
402 cout << endl;
403 cout << "Enter file name for key121";
404 cin >> fname125;
405 cout << endl;
406 cout << "Enter file name for key122";
407 cin >> fname126;
408 cout << endl;
409 cout << "Enter file name for key123";
410 cin >> fname127;
411 cout << endl;
412 cout << "Enter file name for key124";
413 cin >> fname128;
414 cout << endl;
415 cout << "Enter file name for key125";
416 cin >> fname129;
417 cout << endl;
418 cout << "Enter file name for key126";
419 cin >> fname130;
420 cout << endl;
421 cout << "Enter file name for key127";
422 cin >> fname131;
423 cout << endl;
424 cout << "Enter file name for key128";
425 cin >> fname132;
426 cout << endl;
427 cout << "Enter file name for key129";
428 cin >> fname133;
429 cout << endl;
430 cout << "Enter file name for key130";
431 cin >> fname134;
432 cout << endl;
433 cout << "Enter file name for key131";
434 cin >> fname135;
435 cout << endl;
436 cout << "Enter file name for key132";
437 cin >> fname136;
438 cout << endl;
439 cout << "Enter file name for key133";
440 cin >> fname137;
441 cout << endl;
442 cout << "Enter file name for key134";
443 cin >> fname138;
444 cout << endl;
445 cout << "Enter file name for key135";
446 cin >> fname139;
447 cout << endl;
448 cout << "Enter file name for key136";
449 cin >> fname140;
450 cout << endl;
451 cout << "Enter file name for key137";
452 cin >> fname141;
453 cout << endl;
454 cout << "Enter file name for key138";
455 cin >> fname142;
456 cout << endl;
457 cout << "Enter file name for key139";
458 cin >> fname143;
459 cout << endl;
460 cout << "Enter file name for key140";
461 cin >> fname144;
462 cout << endl;
463 cout << "Enter file name for key141";
464 cin >> fname145;
465 cout << endl;
466 cout << "Enter file name for key142";
467 cin >> fname146;
468 cout << endl;
469 cout << "Enter file name for key143";
470 cin >> fname147;
471 cout << endl;
472 cout << "Enter file name for key144";
473 cin >> fname148;
474 cout << endl;
475 cout << "Enter file name for key145";
476 cin >> fname149;
477 cout << endl;
478 cout << "Enter file name for key146";
479 cin >> fname150;
480 cout << endl;
481 cout << "Enter file name for key147";
482 cin >> fname151;
483 cout << endl;
484 cout << "Enter file name for key148";
485 cin >> fname152;
486 cout << endl;
487 cout << "Enter file name for key149";
488 cin >> fname153;
489 cout << endl;
490 cout << "Enter file name for key150";
491 cin >> fname154;
492 cout << endl;
493 cout << "Enter file name for key151";
494 cin >> fname155;
495 cout << endl;
496 cout << "Enter file name for key152";
497 cin >> fname156;
498 cout << endl;
499 cout << "Enter file name for key153";
500 cin >> fname157;
501 cout << endl;
502 cout << "Enter file name for key154";
503 cin >> fname158;
504 cout << endl;
505 cout << "Enter file name for key155";
506 cin >> fname159;
507 cout << endl;
508 cout << "Enter file name for key156";
509 cin >> fname160;
510 cout << endl;
511 cout << "Enter file name for key157";
512 cin >> fname161;
513 cout << endl;
514 cout << "Enter file name for key158";
515 cin >> fname162;
516 cout << endl;
517 cout << "Enter file name for key159";
518 cin >> fname163;
519 cout << endl;
520 cout << "Enter file name for key160";
521 cin >> fname164;
522 cout << endl;
523 cout << "Enter file name for key161";
524 cin >> fname165;
525 cout << endl;
526 cout << "Enter file name for key162";
527 cin >> fname166;
528 cout << endl;
529 cout << "Enter file name for key163";
530 cin >> fname167;
531 cout << endl;
532 cout << "Enter file name for key164";
533 cin >> fname168;
534 cout << endl;
535 cout << "Enter file name for key165";
536 cin >> fname169;
537 cout << endl;
538 cout << "Enter file name for key166";
539 cin >> fname170;
540 cout << endl;
541 cout << "Enter file name for key167";
542 cin >> fname171;
543 cout << endl;
544 cout << "Enter file name for key168";
545 cin >> fname172;
546 cout << endl;
547 cout << "Enter file name for key169";
548 cin >> fname173;
549 cout << endl;
550 cout << "Enter file name for key170";
551 cin >> fname174;
552 cout << endl;
553 cout << "Enter file name for key171";
554 cin >> fname175;
555 cout << endl;
556 cout << "Enter file name for key172";
557 cin >> fname176;
558 cout << endl;
559 cout << "Enter file name for key173";
560 cin >> fname177;
561 cout << endl;
562 cout << "Enter file name for key174";
563 cin >> fname178;
564 cout << endl;
565 cout << "Enter file name for key175";
566 cin >> fname179;
567 cout << endl;
568 cout << "Enter file name for key176";
569 cin >> fname180;
570 cout << endl;
571 cout << "Enter file name for key177";
572 cin >> fname181;
573 cout << endl;
574 cout << "Enter file name for key178";
575 cin >> fname182;
576 cout << endl;
577 cout << "Enter file name for key179";
578 cin >> fname183;
579 cout << endl;
580 cout << "Enter file name for key180";
581 cin >> fname184;
582 cout << endl;
583 cout << "Enter file name for key181";
584 cin >> fname185;
585 cout << endl;
586 cout << "Enter file name for key182";
587 cin >> fname188;
588 cout << endl;
589 cout << "Enter file name for key183";
590 cin >> fname189;
591 cout << endl;
592 cout << "Enter file name for key184";
593 cin >> fname190;
594 cout << endl;
595 cout << "Enter file name for key185";
596 cin >> fname191;
597 cout << endl;
598 cout << "Enter file name for key186";
599 cin >> fname192;
600
```

```

1 * Start developing or writing
private < ifstream.h>
FPS<close(); read();> > stream file
FPT - close() < ifstream.h> > > stream file
FPS. open (frame); < ifstream.h> > > stream file
if (!FPS)
 {
 cout<<"in Error! In opening source file";
 cout<<"In press any key to exit...";>> cin;
 getch();
 exit(1);
 }
 if (control) {cout << "File";
 FPT .open ("temp .txt");
 if (!FPT)
 {
 cout<<"In opening file temp";
 exit(1);
 }
 cout<<"Close file";
 FPT .close();
 cout<<"In press any key to exit...";>> cin;
 getch();
 exit(1);
}
 {
 cout<<"In write mode";>> cin;
 cout<<"In press any key to exit...";>> cin;
 getch();
 exit(1);
 }
 {
 cout<<"In read mode";>> cin;
 cout<<"In press any key to exit...";>> cin;
 getch();
 exit(1);
 }
 {
 cout<<"In append mode";>> cin;
 cout<<"In press any key to exit...";>> cin;
 getch();
 exit(1);
 }
 {
 cout<<"In binary mode";>> cin;
 cout<<"In press any key to exit...";>> cin;
 getch();
 exit(1);
 }
 {
 cout<<"In text mode";>> cin;
 cout<<"In press any key to exit...";>> cin;
 getch();
 exit(1);
 }
}
 {
 cout<<"In File<< frame<< encrypted spectrum";
 cout <<"In File<< frame<< encrypted spectrum";
 FPS<close();>> > > stream file
 FPT .close();
 FPS .close();
 getch();
 cout <<"In press any key to exit...";>> cin;
 getch();
 exit(1);
 }
}

```

```

FPS->close();
cout << "File successfully closed" << endl;
FPT->close();
cout << "File successfully closed" << endl;
FPS->open(Fname);
if (!FPS)
{
 cout << "Error in opening file" << endl;
 exit(1);
}
FPT->open("temp.txt");
if (!FPT)
{
 cout << "Error in opening file temp" << endl;
 FPT->close();
 cout << "In press any key to exit." << endl;
 getch();
 exit(4);
}
while (FPT->eof() != true)
{
 cout << FPT->getchar();
}
FPT->close();
cout << "File successfully closed" << endl;
cout << "File successfully encrypted successfully" << endl;
cout << "In pressing key to exit" << endl;
FPS->close();
FPT->close();
cout << "File successfully closed" << endl;
cout << "File successfully closed" << endl;

```

3. Write a program to display file's content.

```
#include <iostream.h>
#include <fstream.h>
void main()
{
 char ch;
 ifstream fin;
 ofstream fout;
 fin.open("sample.txt");
 cout << "Hello";
 fin.close();
 cout << endl;
 cout << "File contents are : ";
 fin.open("sample.txt");
 while (!fin.eof())
 {
 fin.read((char *) &ch, 1);
 cout << ch;
 }
 fin.close();
}
```

@SumitBiranje