

OOPS

Experiment 1(a)

- Aim : To study header files in C++ & give a detailed description.
- Software Used : Turbo C.

(i)

`<iostream.h>`

This header file declares the basis of C++ stream input/output routines. Some of the functions defined in it are open, close, get, read, write etc.

(ii)

`<stdio.h>`

This header file defines the type of macros needed for the standard I/p packages. This file also defines the standard I/O predefined streams, i.e. `stdin`, `stdout`.

(iii)

`<String.h>`

This header file declares several string manipulation & memory manipulation routines. The functions contained in `string.h` are -
`memchr`, `memcpy`, `strcmpi`, `memdate`, `memset`,
`strcat`, `strcmp`.

(4). <math.h>

This header file declares prototypes for the math functions & math error handlers. The routines in math.h file perform mathematical calculations & conversions. Files included are - div, exp, log, sin, cos, fabs etc.

(5). <Stdlib.h>

This header file declares several commonly used routines like search / sort etc. Some of its functions are - fullpath, exit, free, malloc, realloc etc.

(6). <iomanip.h>

This header file declares the C++ streams I/O manipulators & contains macros for creating parameterized manipulators. Some of its functions are -

dec, endl, flush, hex, oct, setw, ws etc.

Experiment 1 (b)

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

```
struct student
```

```
{  
    int rollno;  
    char name[25];  
};
```

```
void main()
```

```
{  
    student s1, s2;  
    clrscr();  
    cout << "Enter roll no. & name : ";  
    cin >> s1.rollno >> s1.name;  
    cin >> s2.rollno >> s2.name;  
    cout << "The roll no. & names are as follows - ";  
    cout << s1.rollno << s1.name;  
    cout << s2.rollno << s2.name;  
    getch();
```

```
}
```

Jahid
12/9/12

Experiment 2

- Aim : WAP to implement an account class with member function -
 - (i) To compute interest
 - (ii) Show balance.
 - (iii) Withdraw
 - (iv) Deposit
- Software Used : Turbo C++ .
- Source Code

```
#include <iostream.h>
#include <stdlib.h>
#include <math.h>
#include <process.h>
class account
{
    int deposit, withdraw, rate, time;
    float balance, si;
public:
    account()
    {
        balance = 0; withdraw = 0;
        deposit = 0; rate = 0;
        time = 0; si = 0;
    }
}
```

```
void deposit()
```

{

```
balance + = si;
```

```
cout << "In Enter the amount to be deposited : ";
```

```
cin >> deposit;
```

```
balance + = deposit;
```

{

```
void draw()
```

{

```
balance + = si;
```

```
cout << "Enter the amount you wish to withdraw : ";
```

```
cin >> withdraw;
```

{

```
if (balance >= withdraw)
```

```
{ balance - = withdraw;
```

{

```
else {
```

```
cout << "Not enough balance!" ;
```

{

}

```
void interest()
```

{

```
si = (balance * rate * time) / 1000;
```

```
cout << "Your balance without interest : " << balance;
```

```
cout << "Simple interest on your balance : " << si ;
```

```
balance + = si;
```

```
cout << "Total balance : " << balance ;
```

{

```
void input ()  
{  
    cout << "In Enter your amount : ";  
    cin >> balance;  
    cout << "In Enter rate of interest /year : ";  
    cin >> rate;  
    cout << "In Enter time : ";  
    cin >> time;  
}  
void show ()  
{  
    interest (); };
```

```
void main () {  
    clrscr();  
    account a;  
    a.input ();  
    int ch; label;  
    cout << "In Enter your choice : In 1. Deposit  
    \n 2. Withdraw \n 3. Show \n 4. Exit";  
    cin >> ch;  
    switch (ch)  
    {  
        case 1 : a.deposit (); break;  
        case 2 : a.withdraw (); break;  
        case 3 : a.show (); break;  
        case 4 : exit(0); -}  
    goto label; getch();  
}
```

Sakshi (10)
12/9/18

Experiment 3

- Aim : Find the greatest of 2 numbers using 2 different classes & using friend function.
- Software Used : Turbo C

- Source Code

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

class b;
class a
{
private :
    int m ;
public :
    void setvalue (int i)
    { m = i ; }
    friend void max (a, b);
};

class b
{
int n ;
public :
    void setvalue (int j)
    { n = j ; }
```

```
friend void max(a, b);  
};  
void max (aa1, bb1)  
{  
    if((a1.m) > (b1.n))  
        cout << "Greatest is :" << a1.m;  
    else  
        cout << "Greatest is :" << b1.n;  
}  
  
void main()  
{  
    class();  
    a a2;  
    b b2;  
    a2.setvalue(100);  
    b2.setvalue(200);  
    max(a2, b2);  
    getch();  
}
```

~~Abhi~~
19/2/18 (13)

Experiment 4

- Aim : Write a program -
Imagine a ticket selling booth at a fair.
People passing by are requested to purchase a ticket. A ticket is priced at 250/- . The booth keeps track of the no. of people visiting the booth and the total amount of money collected.

Model this ticket selling booth with a class ticketbooth including following members -

Data Members

- No. of people visited
- Total amount of money collected.

Member functions

- To assign initial values (assign 0 to both data members).
- To increment only people in case ticket is not sold out.
- To increment people total as well as amount total if a ticket is sold.
- To display the two totals.
- To display the no. of tickets sold out.

(contd.)

- Software Used : Turbo C

- Source Code

```
#include <iostream.h>
#include <conio.h>
class ticketbooth
{
private :
    float amount ;
    int npv, c, y ;
public :
    void initialise()
    {
        npv = 0 ;
        amount = 0.0 ;
        c = 0 ;
    }
    void increment()
    {
        npv++;
        c++ ;
    }
    void increment2C()
    {
        npv++;
        amount = amount + 2.5 ;
    }
}
```

```
void show ()  
{  
    y = npv - c;  
    cout << "Total people visited : " << npv << endl;  
    cout << "Total amount collected : " << amount << endl;  
    cout << "Total tickets sold : " << y << endl;  
}  
};  
void main ()  
{  
    clrscr ();  
    int k;  
    ticketbooth o;  
    o. initialize ();  
    while (k != 0)  
    {  
        cout << "Enter 1 if person visited but ticket not sold : ";  
        cout << "Enter 2 if ticket is sold : " << endl;  
        cout << "Enter 0 to exit. " << endl;  
        cin >> k;  
        switch (k)  
        {  
            case 1: o. increment ();  
                o. show (); break;  
            case 2:  
                o. increment2 ();  
                o. show (); break;  
            default: cout << "EXIT";  
                break; } }  
getch();
```

lalwli
12/2/19
14

Page No.	
Date	

Experiment 5

- Aim : WAP to find out the no. of members in a class using a static member function.
- Software Used : Turbo C++.
- Software Used = Source Code

```
#include <iostream.h>
#include <conio.h>
class student {
    int roll;
    static int count;
public:
    void setroll()
    {
        roll++;
        count++;
    }
    static void showroll()
    {
        cout << "No. of objects :" << count;
    }
};

int student::count;
void main()
{}
```

Page No.	
Date	

```
Student s1, s2, s3, s4;  
clrscr();  
s1.setroll();  
s2.setroll();  
s3.setroll();  
s4.setroll();  
student :: showroll();  
getch();
```

{

~~Aakash
12/9/12 (B)~~

Experiment 6

- Aim : To print the sum of two complex numbers using function ^{constructor} overloading .
- Software Used : Turbo C++ .
- Source Code

```
#include <iostream.h>
#include <conio.h>
class complex {
public :
    int a, b;
    complex (int e=0) {
        a = 5;
        b = 6;
        cout << " If you don't provide values , "
            " initial values will be taken " << a << "&" << b << endl;
    }
    complex (int m, int n) {
        cout << " You've called a constructor with both "
            " int values " << m << " and " << n << endl;
    }
    complex (int m, float n) {
```

Output

Sum is $1966 + 19.36j$

Sum is $5 + 5j$

Sum is $5 + 76j$

cout << " You called a constructor with two diff. data types, one int & other float" << m << " and " << endl;

}

}

void main () {

int a, b;

float c, d;

a = 5;

b = 6;

c = 5.80;

d = 6.998;

chase();

consver first (d);

consver second (a, b);

consver third (a, c);

getch();

}

Abhi
5/3/18 (13)

Experiment 7

- Aim : To implement a class string containing 3 overloaded operators & functions to display length of string, Convert lower case to upper case & vice versa.

- Software Used : Turbo C++

- Source Code

```
#include <iostream.h>
#include <string.h>
#include <conio.h>
class my_string
{
    char str[20]
public :
    void getdate();
    void display();
    int operator=(my_string str1);
    void operator+(my_string str1);
    void operator<=(my_string str1);
    void up();
    void low();
    void l();
};
void my_string::getchar()
```

Page No.	
Date	

```

{
    cout << "Enter the string : " << endl;
    cin >> str;
}

void my_string::display()
{
    cout << str << endl;
}

int my_string::operator=(my_string str1)
{
    strcpy(str, str1.str);
    cout << "String after copying is : = " << str;
}

void my_string::operator+(my_string str1)
{
    strcat(str, str1.str);
    cout << "String after concatenation is : " << str;
}

void my_string::operator<=(my_string str1)
{
    if (strcmp(str, str1.str) == 0)
        cout << "Both strings are equal" << endl;
    else
        cout << "Both strings are not equal!" << endl;
}

void l()
{
    cout << strlen(str) << endl;
}

void up()
{
   strup(str);
}

```

Output

Enter the string : Delhi

Enter the string : Mumbai

Delhi
Delhi Mumbai
The strings are not equal

5

Delhi

DELHI

6

MUMBAI

Mumbai

cout << str << endl; }

void low()

{ strlow(str);

cout << str << endl;

}

void manh()

{

clrscr()

mystring a, b;

a.getdata();

b.getdata();

b = a;

a+b;

a <= b;

a.l();

a.up();

a.low();

b.l();

~~b.up(); b.low();~~

getch();

}

Delhi
9/4/13. 11

Page No.	
Date	

Experiment - 8

- Aim: Create 3 classes - Student that stores student roll-no., test that stores the marks obtained in 2 subjects & result that contains the total marks obtained by the student. Class sport stores weightage for sport that the student opts. Class Result calculates the total marks considering sports weightage also & displays the result. Create main() function that instantiate the result class object & displays the final result of a student.

- S/w Used: Turbo C++.

- Source Code

```
#include <iostream.h>
class student
{
public:
    int rno;
    void getdata()
    {
        cout << "Enter roll no " << endl;
        cin >> rno;
    }
    void display()
    {
        cout << "The roll no is : " << rno;
    }
};
```

Page No.	
Date	

```

class test : public student
{
public :
    int m1, m2;
    void getData2()
    {
        cout << "Enter marks of the students : " << endl;
        cin >> m1 >> m2;
    }
    void display2()
    {
        cout << "The marks of the student are : " << endl;
        cout << m1 << m2;
    }
};

class sport()
{
public :
    int smark;
    void getData3()
    {
        cout << "Enter marks of the students for sports : " << endl;
        cin >> smark;
    }
    void display3()
    {
        cout << "The marks for sports : ";
        cout << smark; } } ;

```

Output

Roll no. : 12
 Total marks : 220

```

class result : public test, public sport
{
    int total;
public:
    void display()
    {
        int t;
        t = m1 + m2 + smark;
        cout << "Total marks : " << endl;
        cout << t;
    }
    void main()
    {
        clrscr();
        result s;
        s.getdata1();
        s.getdata2();
        s.getdata3();
        s.getdata4();
        s.display();
        getch();
    }
}

```

(1) 11/11/18

(13)