

| Sr. No. | Name of the Program |
|---------------------|---|
| C++ Programs | |
| 1 | Bubble Sort. |
| 2 | Call by Reference using Swap function. |
| 3 | Implementation of Binary Search. |
| 4 | Traversing an array with a pointer. |
| 5 | Reversal of a string. |
| 6 | Reversal of a link list. |
| 7 | Implementation of member function using Scope Resolution Operator (::). |
| 8 | Implementation of circle class using a Constructor. |
| 9 | Implementation of ratio class using a Constructor. |
| 10 | Implementation of Constructor & Destructor. |
| 11 | Implementation of Complex Constructor. |
| 12 | Implementation of '+' & '/' operator overloading. |
| 13 | Implementation of Hybrid Inheritance. |
| 14 | Implementation of Simple Virtual Function. |
| 15 | Creation of a text file. |

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For

Diagram _____

Obs. Table _____

Calculations _____

Graphs _____

Results _____

Unit _____

Name VRUSHIL SONI.

Class XII Roll No. 12032

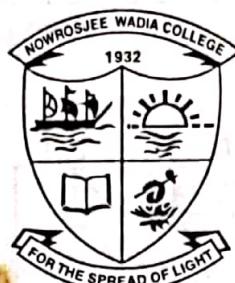
Submitted On 22/7/15 29.7.15

Performed On 22/7/15

Signature

Experiment Complete

Incharge



Expt. No. 1 Title Sorting numbers in ascending & descending order by bubble sort.

Aim: Write a program in C++ that initializes an array of 10 real numbers. The program must sort numbers in ascending & descending order using bubble sort method. Print the given list is sorted & unsorted form.

Algorithm:-

Step 1: START

2: Initialize array a[10] with 10 values.

3: Initialize variables required in program.

4: Print a[10] in unsorted Order.

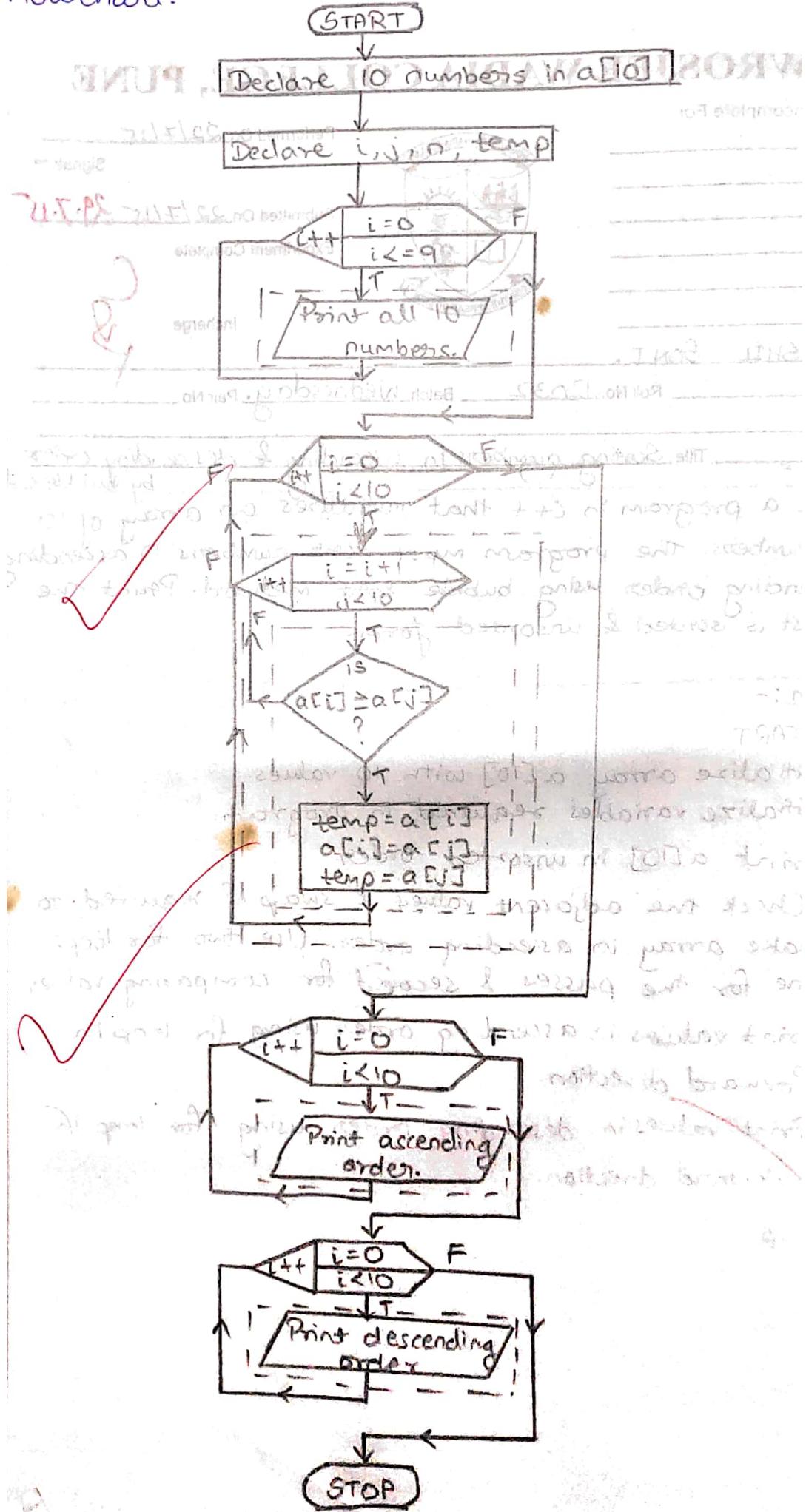
5: - Check the adjacent values & swap if required. to make array in ascending order. Use two for loops: one for the passes & second for comparing values.

6: Print values in ascending order using for loop in forward direction.

7: - Print values in descending order using for loop in backward direction.

8: Stop.

Flowchart:-



BUBBLE.CPP

```
//program performed by roll no 12032
//program performed by using bubble sort
#include<iostream.h>
#include<conio.h>
void main()
{
clrscr();
int a [10]={1,2,3,4,5,6,7,8,9,10};
int i,j,temp;
for(i=0;i<10;i++) //printing original elements
cout<<"the "<<i+1<<" element is "<<a[i]<<endl;
for(i=0;i<10;i++) //no of passes
{
for(int j=i+1;j<10;j++) //comparison
{
if(a[i]>a[j]) //swaping
{
temp=a[i];
a[i]=a[j];
a[j]=temp;
}
}
}
for(i=0;i<10;i++)//printing ascending order
cout<<"ascending order of array is "<<a[i]<<endl;
for(j=9;j>=0;j--)//printing descending order
cout<<"descending order of array is "<<a[j]<<endl;
getch();
}

/* the1element is 1
the2element is 2
the3element is 3
the4element is 4
the5element is 5
the6element is 6
the7element is 7
the8element is 8
the9element is 9
the10element is 10
ascending order of array is=1
ascending order of array is=2
ascending order of array is=3
ascending order of array is=4
ascending order of array is=5
```

BUBBLE.CPP

```
ascending order of array is=6
ascending order of array is=7
ascending order of array is=8
ascending order of array is=9
ascending order of array is=10
descending order of array is=10
descending order of array is=9
descending order of array is=8
descending order of array is=7
descending order of array is=6
descending order of array is=5
descending order of array is=4
descending order of array is=3
descending order of array is=2
descending order of array is=1 */

```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For

Diagram _____

Obs. Table _____

Calculations _____

Graphs _____

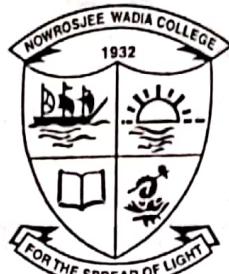
Results _____

Unit _____

Name Vrushil Soni.

Class XII Sci.

Roll No. 12032 Batch Wednesday. Pair No. _____



Performed On 29/7/15.

Signature

Submitted On 5.7.15

Experiment Complete

Incharge G

Expt. No. 2 Title Use of Swap Function.

Aim: Write a program in C++ which swaps or exchanges data using 3 swapping methods (pass by value, pass by reference, pass by address pointer) using switch case.

Program Analysis:-

Step 1: Start.

Step 2: Declare variables ch, x, y, n.

Step 3: Declare swap function.

Step 4: Create a do loop.

Step 5: Print "which swap method do u want to choose"

Step 6: switch n.

Step 7: If n=1, then s1(x,y) and s1(int a, int b)

Step 8: int temp=a

a=b

b=temp.

Step 9: Return & Print swapped values & then break.

Step 10: If n=2, then s2(x,y) & s2(int &a, int &b).

Step 11: int temp=a

a=&b

b=temp.

Step 12: Return & Print swapped values & then break.

Step 13: If n=3, then s3(&x,&y) & s3(int *a, int *b).

Step 14: int temp=*a

*a=&b

*b=temp.

Step 15: Return & Print swapped values & then break.

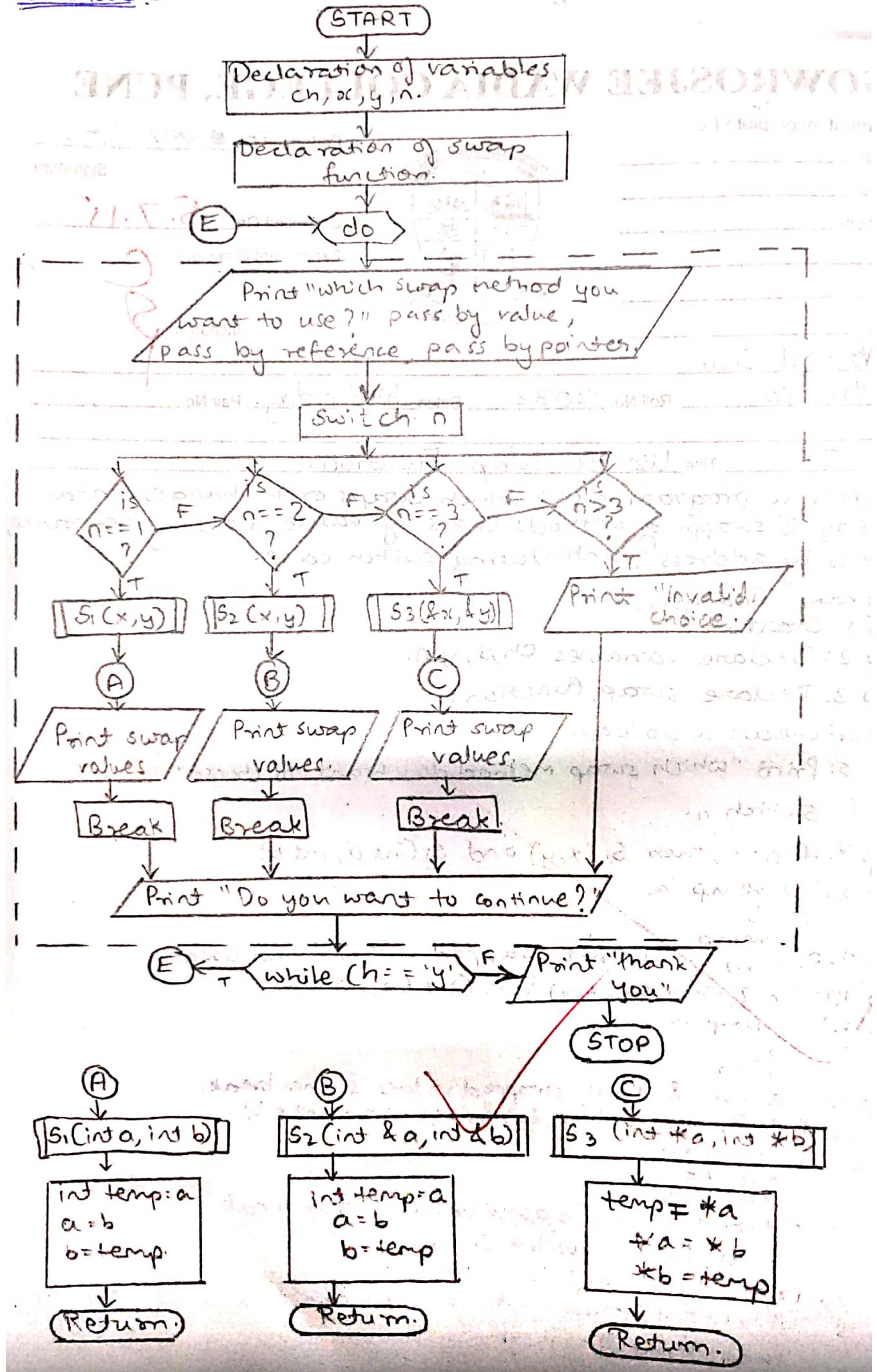
Step 16: If n=31, Print "Invalid choice".

Step 17: Print "do u want to continue?"

Step 18: If ch==y, then go to do loop again & if ch is not, then Print "thank you".

Step 19: STOP.

Flowchart :-



```
SWAPPING.CPP
//program performed by 12032
//program which swaps or exchanges data using 3
swapping methods
#include<iostream.h>
#include<conio.h>
void main()
{
    clrscr();
    char ch='y';
    int x,y;
    int n;
    void s1(int,int);
    void s2(int&,int&);
    void s3(int*,int*);
    do
    {
        cout<<"which swapping method u want to
use?"<<endl;
        cout<<"1:pass by value\n2:pass by reference \n3:
pass by address or pointer using switch
case"<<endl;
        cin>>n;
        switch(n)
        {
            case 1:
            {
                cout<<"enter 2 nos"<<endl;
                cin>>x>>y;
                s1(x,y);
                cout<<"the swaped nos are"<<x<<"&"<<y<<endl;
                break;
            }
            case 2:
            {
                cout<<"enter the nos to be swapped"<<endl;
                cin>>x>>y;
                s2(x,y);
                cout<<"the swapped nos. are"<<x<<"&"<<y<<endl;
            }
        }
    } while(ch=='y');
}
```

SWAPPING.CPP

```
break;
}
case 3:
{
    cout<<"enter the nos to be swapped" << endl;
    cin>>x>>y;
    s3(&x,&y);
    cout<<"the swapped values
are" <<x <<"&" <<y << endl;
    break;
}
default:
{
    cout<<"please enter a valid value" << endl;
}
}
cout<<"do u want to continue?" << endl;
cin>>ch;
}
while (ch=='y');
{
    cout<<"thank you" << endl;
}
getch();
}
void s1(int a,int b)
{
    int temp=a;
    a=b;
    b=temp;
}
void s2(int&a,int&b)
{
    int temp=a;
    a=b;
    b=temp;
}
void s3(int*a,int*b)
```

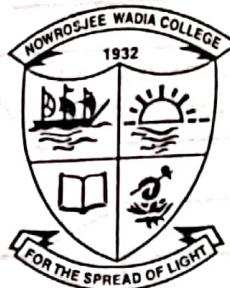
```
{  
    int temp=*a;  
    *a=*b;  
    *b=temp;  
};  
/*the swaped nos are2&4  
do u want to continue?  
y  
which swapping method u want to use?  
1:pass by value  
2:pass by reference  
3: pass by address or pointer using switch case  
2  
enter the nos to be swapped  
5 7  
the swapped nos. are7&5  
do u want to continue?  
y  
which swapping method u want to use?  
1:pass by value  
2:pass by reference  
3: pass by address or pointer using switch case  
3  
enter the nos to be swapped  
8 5  
the swapped values are5&8  
do u want to continue?  
n  
thank you*/
```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____
Obs. Table _____
Calculations _____
Graphs _____
Results _____
Unit _____

Performed On 5/8/15

Signature



Submitted On 19-8-15

Experiment Complete

Incharge

Name Vrushil Soni Class XII Sci. Roll No. 12032 Batch Wednesday Pair No. -

Expt. No. 3 Title Binary Search.

Aim: Write a program in C++ which initializes array of 10 real numbers, take numbers in sorted manner & then search location of no. using binary search method.

Program Analysis.

Step 1: Start.

Step 2: Initialize arrays & declare variables.

Step 3: Print "numbers in sorted order are".

Step 4: for i=0, i<=9, i++, print a[i]. If false, then search for any no. in array.

Step 5: Read item

Step 6: Use formula mid = (beg + end) / 2.

Step 7: Create while loop: (beg <= end), if true, then is item == a[mid]?

If false then go to step 10. If true, then x = mid, n = 1.

Step 8: Break & go to step 12.

Step 9: If while loop is false, then go to step 12.

Step 10: Check if item > a[mid]

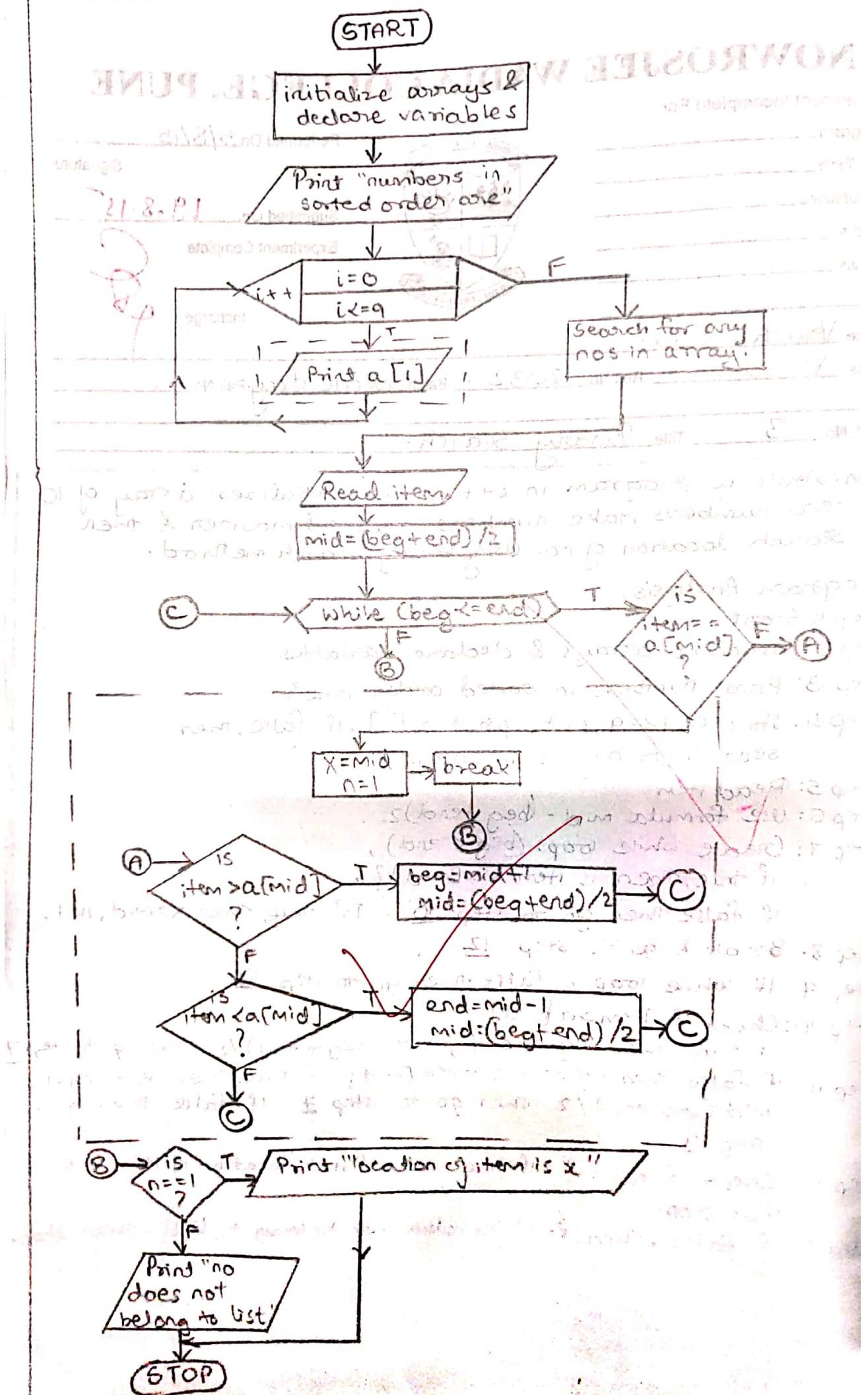
If true then beg = mid + 1, mid = (beg + end) / 2. Then go to step 7

Step 11: If false then check if item < a[mid]. If true then end = m - 1, mid = (beg + end) / 2 then go to step 7. If false then go to step 7.

Step 12: Check if n == 1? If true then Print "location of item is x" & then stop.

Step 13: If false, then Print "no does not belong to list". then stop.

Flowchart :-



```

//program performed by 12032           BINARYSE.CPP
#include<iostream.h>
#include<conio.h>
void main()
{
clrscr();
int a[10]={3,7,12,16,21,25,30,34,43,52}; //array
initialization
int mid,i,item,beg=0,end=9,n=0,x;
cout<<"the nos in sorted order are"<<endl;
for(i=0;i<=9;i++)
{
cout<<a[i]<<endl;
}
cout<<"search for any no from the above array"<<endl;
cin>>item;
mid=int((beg+end)/2); //finds midpoint
while(beg<=end)
{
if(item==a[mid])
{
x=mid;
n=1;
break;
}
else if(item>a[mid])
{
beg=mid+1;
mid=int((beg+end)/2);
}
else if(item<a[mid])
{
end=mid-1;
mid=int((beg+end)/2);
}
}
if(n==1)
cout<<"the location of the item is a["<<x<<"]"<<endl;
else
cout<<"the no does not belong to the list"<<endl;
getch();
}

```

BINARYSE.CPP

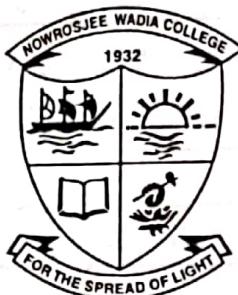
```
1  
/*the nos in sorted order are  
3  
7  
12  
16  
21  
25  
30  
34  
43  
52  
search for any no from the above array  
74  
the no does not belong to the list*/
```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____
Obs. Table _____
Calculations _____
Graphs _____
Results _____
Unit _____

Performed On 26/08/15 _____

Signature



Submitted On 2.9.15 _____

Experiment Complete

Incharge

G

Name Vrushil Soni.

Class XII Roll No. 12032 Batch Wednesday Pair No. -

Expt. No. 4 Title Traversing an array using pointer.

Aim: Write a program in C++ that traverses an array by initializing an array using pointer & print output

Program Analysis:-

Step 1: Start.

Step 2: Initialize pointer ~~array~~ variable sum & initialize array.

Step 3: Assign ~~P = & arr[0]~~

Step 4: Check if $i=0, i < 5, i++$. If false go to step 8

Step 5: If true, then $sum = sum + arr[i]$

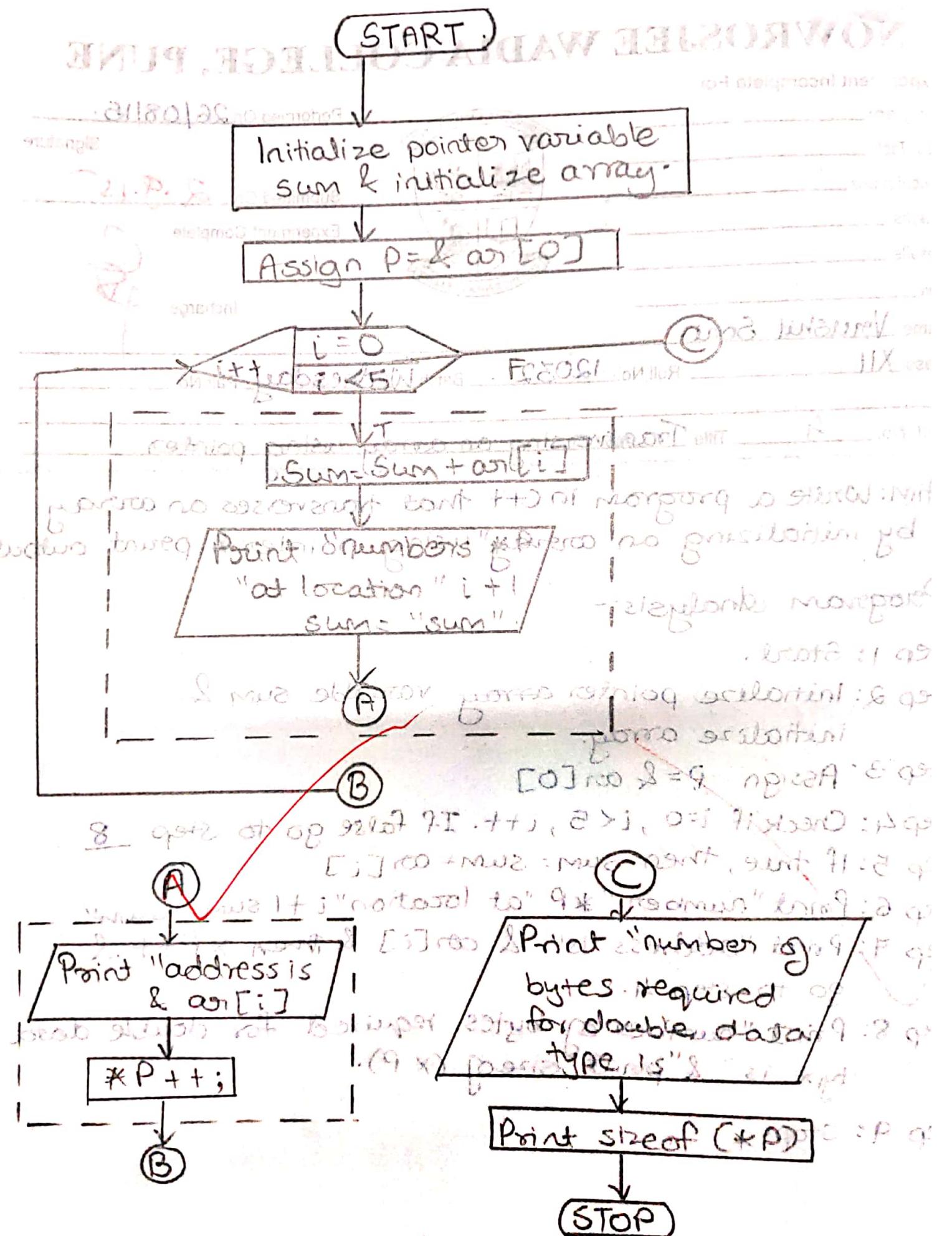
Step 6: Print "number" $*P$ "at location" $i+1$ $sum = "sum"$.

Step 7: Print "address is" & $arr[i]$ & then $*P++$ & go to step 4.

Step 8: Print "number of bytes required for double data type is" & print sizeof ($*P$).

Step 9: Stop.

Flowchart:-



```

//program performed by 12032
//program that traverses an array
#include<iostream.h>
#include<conio.h>
void main()
{
    clrscr();
    double*p,sum,i;
    double ar[5] =
    {2,4,6,8,10};
    p=&ar[0];//assigning address of first element
    for(i=0;i<5;i++)
    {
        sum=sum+ar[i];
        cout<<"number "<<*p<<" at location "<<i+1<<""
        sum= "<<sum<<endl";
        cout<<"address "<<& ar[i]<<endl;
        *p++;
    }
    cout<<"the no. of bytes required for double data
    type pointer is ";
    cout<<sizeof(*p)<<" bytes"<<endl;
    getch();
}
/*number 2 at location 1 sum= 2
address 0x8f94ffbc

number 4 at location 2 sum= 6
address 0x8f94ffc4

number 6 at location 3 sum= 12
address 0x8f94ffcc

number 8 at location 4 sum= 20
address 0x8f94ffd4

```

Page 1

M. E. Society's
NOWROSJEE WADIA COLLEGE, PUNE - 1

TRAVERSE.CPP

number 10 at location 5 sum= 30
address 0x8f94ffd8
the no. of bytes required for double data type
pointer is 8 bytes*/

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For

Diagram _____

Obs. Table _____

Calculations _____

Graphs _____

Results _____

Unit _____

Name VRUSHIL SONI.

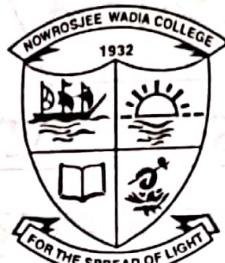
Class XII

Roll No. 12032

Batch Wednesday

Pair No. -

Expt. No. 5 Title String Reverse.



Performed On 19/08/15

Signature

Submitted On 26.8.15

Experiment Complete

Incharge

G

Aim:- Write a program in C++ which reverse the string using function concept. Print original string as well as reverse string.

Program Analysis:-

Step 1: Start.

Step 2: Declare array

char a[50] and int i,j;

Step 3: Put rev (char a[50]).

Step 4: Print "enter a string to be reversed".

Step 5: Read the array a[50]

Step 6: rev (a)

Step 7: Put void rev (char a[50])

Step 8: Declare char array

char b[40] & int i=0, j=0.

Step 9: Create a while loop & check if

a[i] != '\0'. If true then increment i & go to the loop again.

Step 10: If false, then create a while loop & check - -i >= 0.

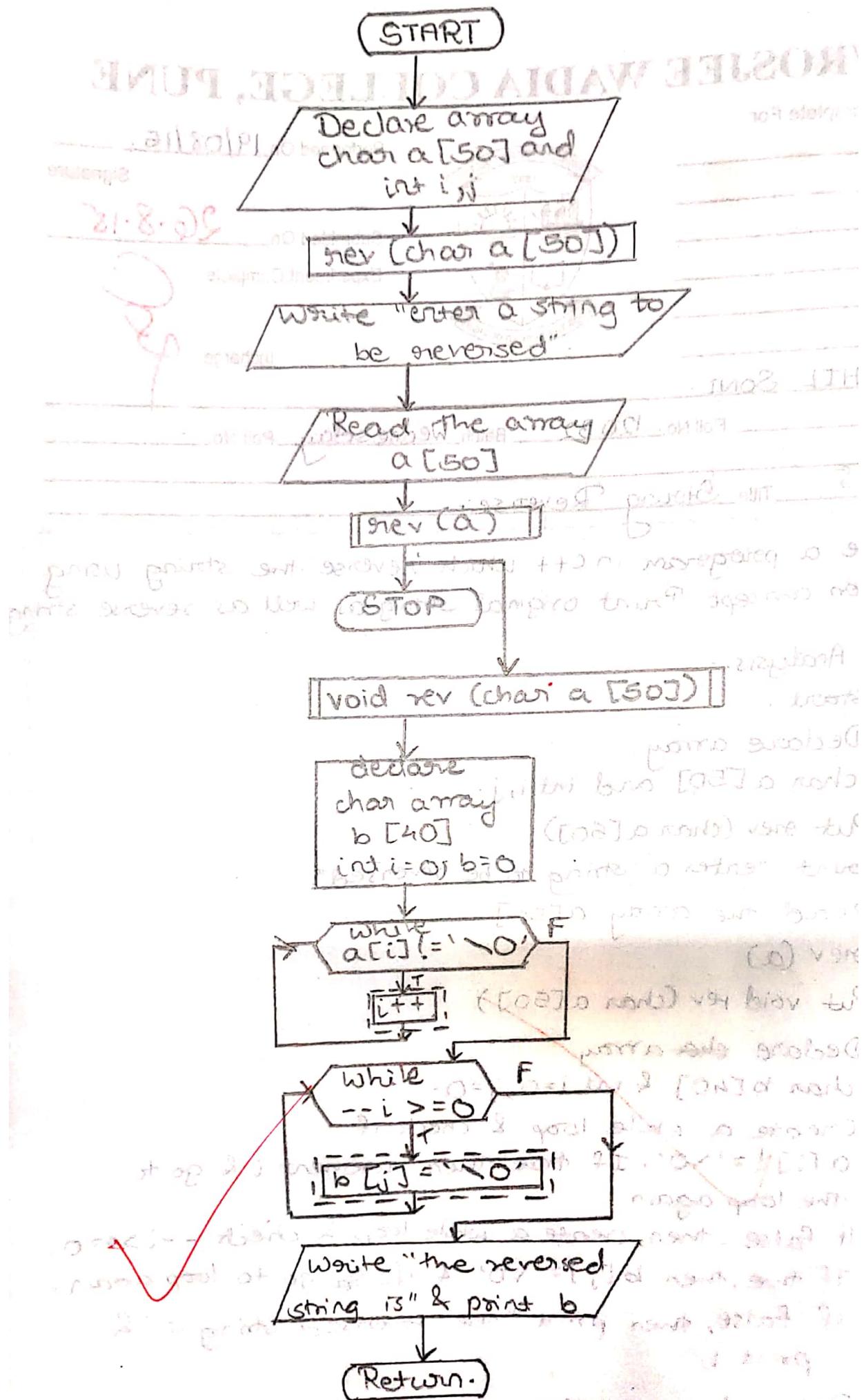
If true, then b[j] = '\0' & ~~if~~ go to loop again.

Step 11: If false, then print "the reversed string is" & print b.

Step 12: Return the program.

Step 13: Stop.

Flowchart :-



```
//program performed by 12032  
//program to reverse a string using function  
concept  
#include<iostream.h>  
#include<conio.h>  
void main()  
{  
    clrscr();  
    char a[50];//array declaration  
    int i,j;  
    void rev(char a[50]);  
    cout<<"enter a string to be reversed"<<endl;  
    cin.getline(a,50);//get string from user  
    rev(a);//function call  
    getch();  
}  
void rev(char a[50])  
{  
    char b[40];  
    int i=0,j=0;  
    while(a[i]!='\0')  
        i++;  
    while(--i>='\0')  
        b[j++]=a[i];  
    b[j]='\0';  
    cout<<"reversed string is"<<endl<<b;  
    getch();  
}  
/*enter a string to be reversed  
abcd  
reversed string is  
dcba*/
```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For

Diagram _____

Obs. Table _____

Calculations _____

Graphs _____

Results _____

Unit _____

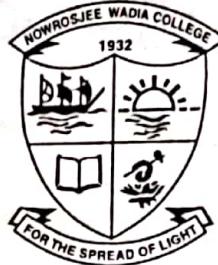
Name Vineeshil Soni.

Class XII

Roll No. 12032

Batch Wednesday

Pair No. —



Performed On 21/10/15

Signature

Submitted On 26/10/15 - 2-11-15

Experiment Complete

Incharge

G

Expt. No. 6. Title Reversal of Link List.

Aim: Write a program in C++ which defines a node class each of whose object contain an integer data member & next pointer. Program should allow user to print linked list in reverse order & traverse list pointing to each data members.

Program Analysis:-

Step 1: Start.

Step 2: Declare link & go to step 8.

Step 3: Use function p.additem(1) & go to step 9.

Step 4: Use fn p.additem(2) & go to step 9.

Step 5: use fn p.additem(3) & go to step 9.

Step 6: Use fn p.additem(4) & go to step 9.

Step 7: Use fn p.display() & go to step 12.

Step 8: Declare link p & then ~~int~~ initialize start=0 & return.

Step 9: Use function void additem (int p) & calculate
node * newlink = newnode.

Step 10: Using pointers declare new → data=p. &
new → next = start.

Step 11: Initialize start = newlink & return.

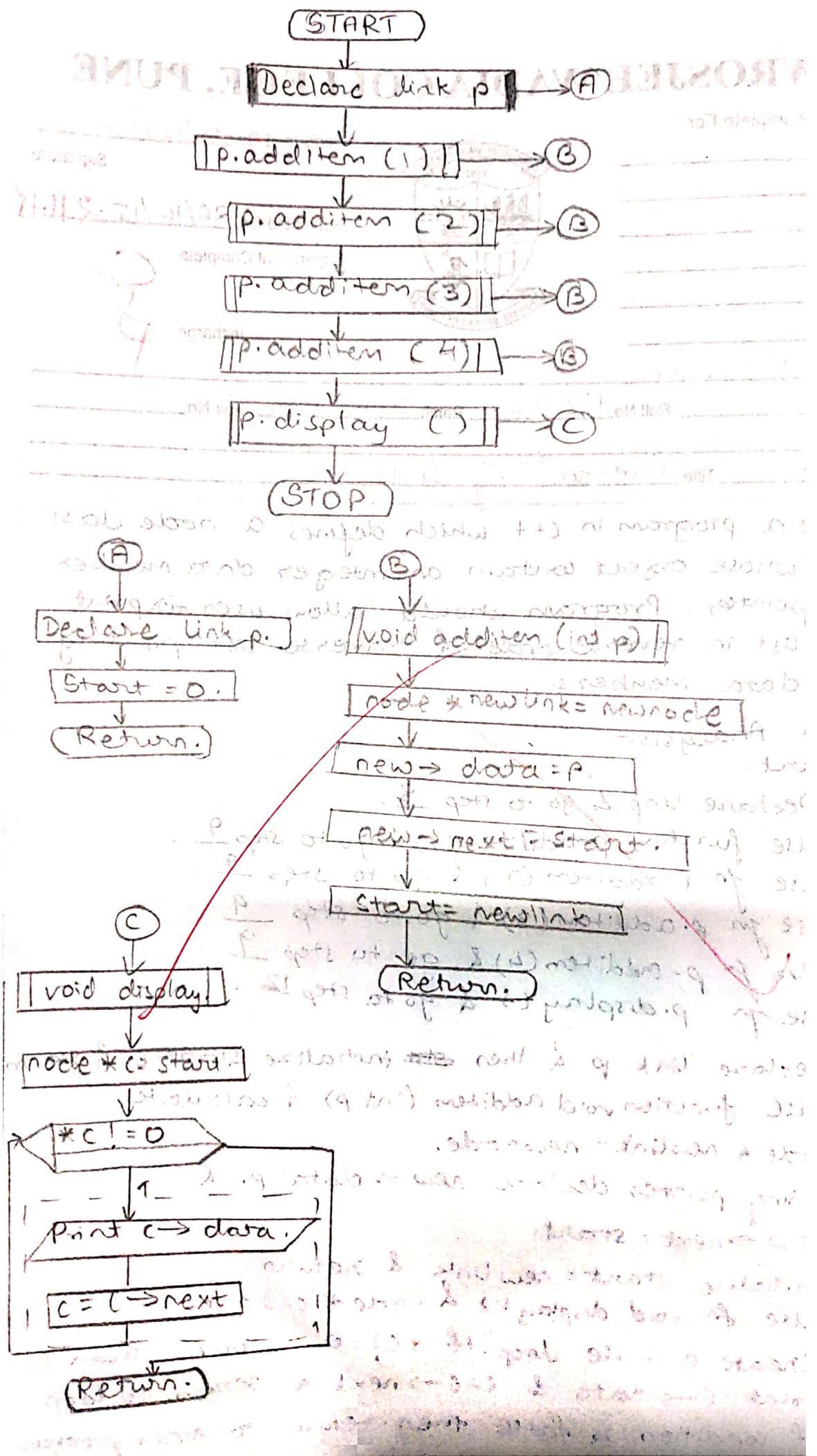
Step 12: Use fn void display() & node * c=start.

Step 13: Create a while loop. If *c!=0 is true then
print c → data & c=c → next & return to loop.

Step 14: If condition is false then return to main program.

Step 15: Stop.

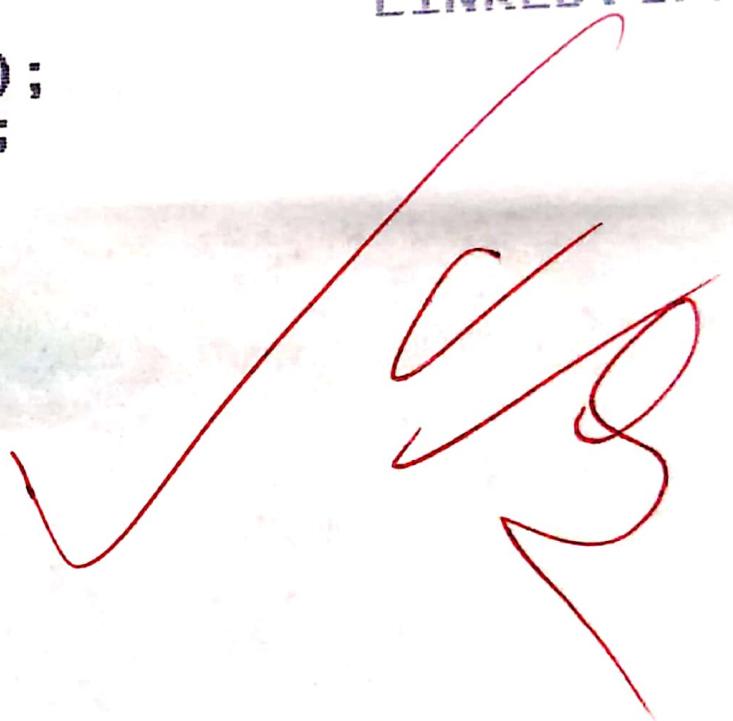
Flowchart:-



```
//program performed by 12032           LINKED.CPP
//program for reversal of linked list
#include<iostream.h>
#include<conio.h>
struct node
{
    int data;
    node *next;
};
class link
{
private:
    node *start;
public:
    link()
    {
        start=0;
    }
    void additem(int p);
    void display();
};
void link::additem(int p)
{
    node *newlink=new node;
    newlink->data=p;
    newlink->next=start;
    start=newlink;
}
void link::display()
{
    node *c=start;
    while(c!=0)
    {
        cout<<c->data<<endl;
        c=c->next;
    }
}
void main()
{
    clrscr();
    link p;
    p.additem(1);
    p.additem(2);
    p.additem(3);
    p.additem(4);
}
```

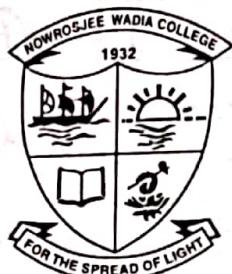
LINKED.CPP

```
p.additem(4);
p.display();
getch();
}
/*4
3
2
1*/
```



NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____
Obs. Table _____
Calculations _____
Graphs _____
Results _____
Unit _____



Performed On 26/8/15 _____ Signature _____

Submitted On 29.15 _____ Experiment Complete _____

Incharge

Name Vrushil Soni

Class XII Roll No. 12032 Batch Wednesday, Pair No. _____

Expt. No. 7 Title Member Jr Implementation

Aim: Write a program in C++ with class ratio using member functions assign () to initialize data members, convert () for double & invert () to create inverse of ratio, print () which prints ratio & its reciprocal.

Program Analysis:-

Step 1: Start.

Step 2: Write ratio ri .

Step 3: Use function ~~ri.assign()~~ & go to step 7.

Step 4: Use function ~~ri.convert()~~ & go to step 8.

Step 5: Use function ~~ri.inverse()~~ & go to step 9.

Step 6: Use function ~~ri.print()~~ & go to step 10.

Step 7: After using the function, ~~go to~~ print "enter numerator & denominator". Read numerator & denominator & then return.

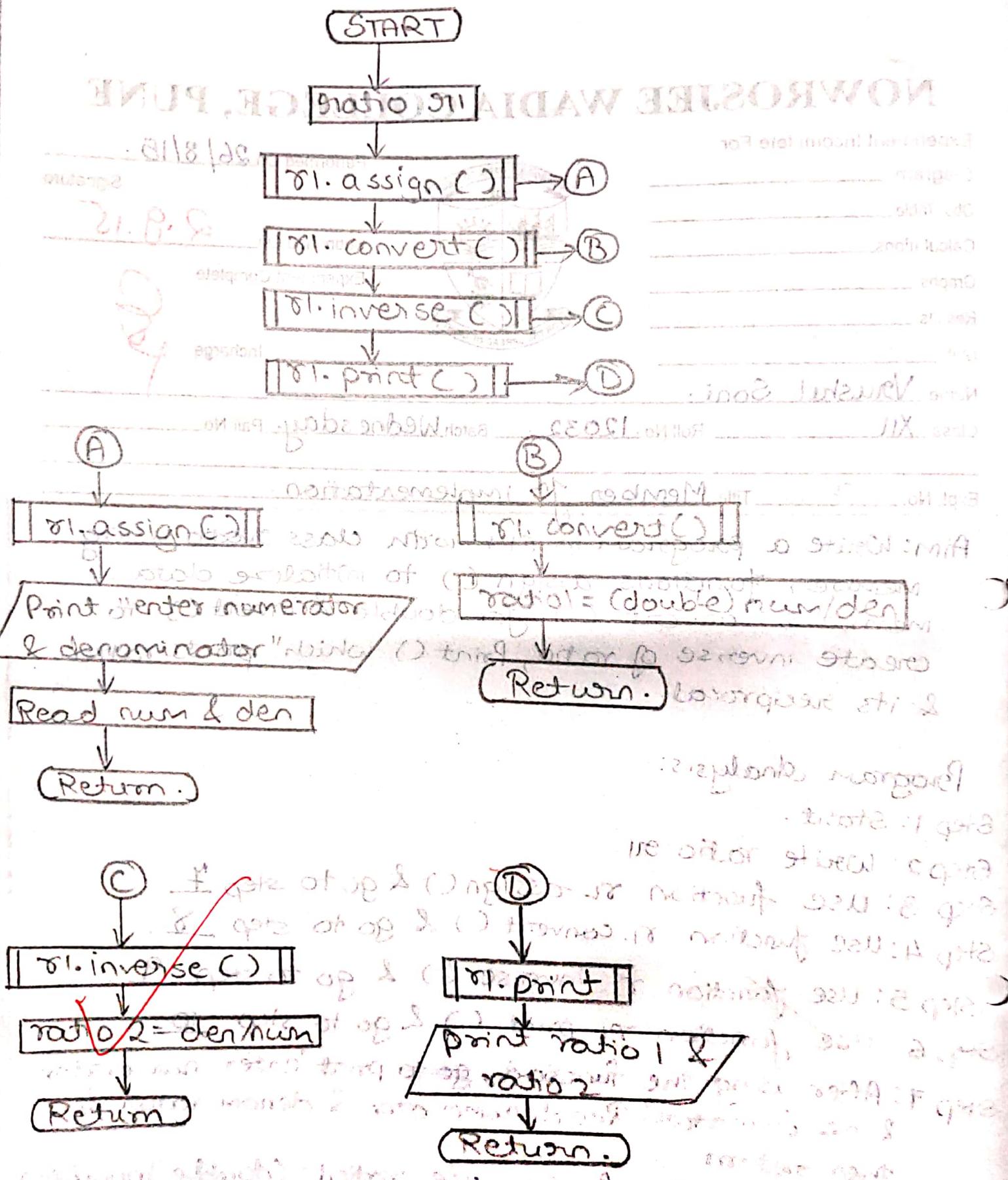
Step 8: After using the function, use $\text{ratio1} = (\text{double})\text{num}/\text{den}$ & then return.

Step 9: After using the function, use $\text{ratio 2} = \text{den}/\text{num}$ & then return.

Step 10: After using the function, ~~the~~ print ratio 1 & ratio 2 & then return.

Step 11: Stop.

Flowchart:-



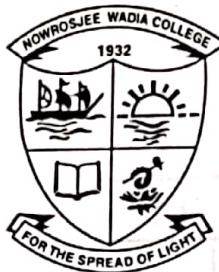
```
//program performed by 12032           MEMFNIMP.CPP
//program to implement member fn
#include<iostream.h>
#include<conio.h>
class ratio
{
private:
int numerator,denominator;
double ratio1,ratio2;
public:
void assign()
{
    cout<<"enter numerator and denominator" << endl;
    cin>>numerator>>denominator;
}
void convert()
{
    ratio1=(double)numerator/denominator;
}
void inverse()
{
    ratio2=(double)denominator/numerator;
}
void print()
{
    cout<<"ratio "<<ratio1<< endl;
    cout<<"inverse ratio "<<ratio2<< endl;
}
void main()
{
clrscr();
ratio r1;
r1.assign();
r1.convert();
r1.inverse();
r1.print();
getch();
}
```

MEMFNIMP.CPP

```
/*enter numerator and denominator  
10  
5  
ratio2  
inverse ratio0.5*/
```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____
Obs. Table _____
Calculations _____
Graphs _____
Results _____
Unit _____



Performed On 16/9/15

Signature _____

Submitted On 23.9.15

Experiment Complete

Incharge C S

Name Vrushil Soni

Class XII Roll No. 12032 Batch WEDNESDAY Pair No. -

Expt. No. 8 Title Implementation of Class Circle using constructor.

Aim: Implement a class circle in C++ where each object will represent a circle storing its radius & x, y co-ordinates of its centre as float. Include a default constructor, member function as area and circumference. The program should print co-ordinates with area & circumference of circle.

Program Analysis:-

Step 1: Start.

Step 2: Print "enter radius" & read as a & print "Enter x & y coordinates" & read as b & c respectively.

Step 3: Implement /use function circle() & go to step 9.

Step 4: Use function circle c1= circle(a,b,c) & go to step 10.

Step 5: Use function circle c3(c2) & go to step 11.

Step 6: Use member fn c1.show("Default circle") & go to step 12.

Step 7: Use member fn c2.show("User defined circle") & go to step 12.

Step 8: Use member junction c3.show("copy circle") & go to step 12.

Step 9: Declare r=7, x=4 & y=6 & return.

Step 10: Use circle function & declare r=a, x=b & y=c & return.

Step 11: Calculate /Declare r= obj.r, x=obj.x, y=obj.y & return.

Step 12: Use function void show (char *msg)

Step 13: Print "msg", "radius" & "x & y coordinates are" x,y

Step 14: Calculate area = (float) 3.145 * pi * pi.

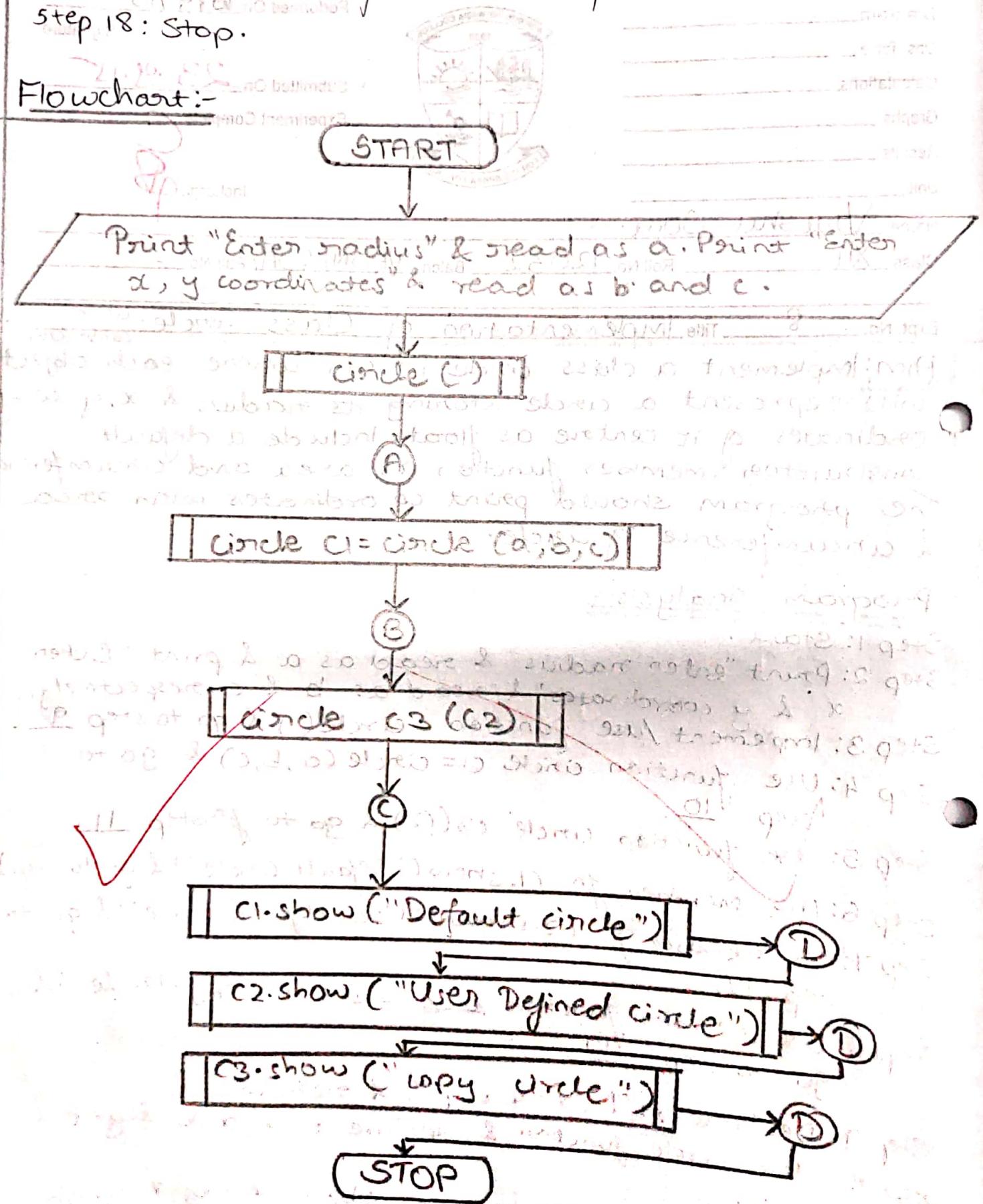
Step 15: Print "Area = " area.

Step 16: Calculate circumference = (float) 2 * 3.145 * pi.

Step 17: Print "Circumference" circumference & return.

Step 18: Stop.

Flowchart:-



A
r=7 , x=4
y=6
Return.

B
circle
r=a
x=b
y=c
Return.

C
r=obj.r
x=obj.x
y=obj.y
Return.

D
void show (char msg)

Print "msg", "radius" r, "x & y coordinates
are " x & y .

area = (float) 3.145 * r * r

Print 'area' area

circ = (float) 2 * 3.145 * r

Print "circumference" circ

Return.

```

CLSCIRC.CPP
//program performed by 12032
//program to implement class circle
#include<iostream.h>
#include<conio.h>
class circle
{
private:
int r;
float area, circ;
int x,y;
public:
circle()// default constructor defination
{
r=7;
x=4;
y=6;
}
circle(float a,float b,float c)//paramaterised
constructor defination
{
r=a;
x=b;
y=c;
}
circle(circle &obj)//copy constructor defination
{
r=obj.r;
x=obj.x;
y=obj.y;
}
void show(char*msg)//show function defination
{
cout<<msg<<endl;
cout<<"radius "<<r<<endl;
cout<<"x and y coordinates are "<<x<<"&"<<y<<endl;
area=(float)3.145*r*r;//calculate area
cout<<"area ="<<area<<endl;
circ=(float)2*3.145*r;//calculate circumference
cout<<"circumference ="<<circ<<endl;
}
};
void main()
{
clrscr();

```

CLSCIRC.CPP

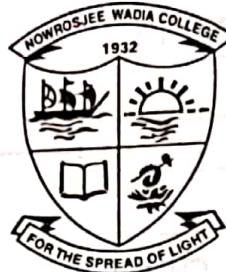
```
float a,b,c;
cout<<"enter radius"<<endl;
cin>>a;
cout<<"enter x coordinate"<<endl;
cout<<"enter y coordinate"<<endl;
cin>>b;
cin>>c;
circle c1;
circle c2(a,b,c);
circle c3(c2);
c1.show("default circle");
c2.show("user inputted circle");
c3.show("copy circle");
getch();
}/*enter radius
1
enter x coordinate
enter y coordinate
8
9
default circle
radius7
x and y coordinates are4&6
area=154.104996
circumference=44.029999
user inputted circle
radius1
x and y coordinates are8&9
area=3.145
circumference=6.29
copy circle
radius1
x and y coordinates are8&9
area=3.145
circumference=6.29*/
```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
 Diagram _____
 Obs. Table _____
 Calculations _____
 Graphs _____
 Results _____
 Unit _____

Performed On 2/9/15

Signature



Submitted On 9.9.15

Experiment Complete

Incharge

Name Vrushil Soni

Class XII Roll No. 12032 Batch Wednesday Pair No. -

Expt. No. do it 9 Title Implementation of Class using default & parameterised constructor

Aim: Write a program in C++ that initializes a ratio class with default constructor & constructor with 1 argument that initializes declare object with default integer value. use display () fn which displays value of ratio.

Program Analysis:-

Step 1: Start

Step 2: Initialize ratio1 & go to step 9.

Step 3: Declare int den.

Step 4: Initialize ratio r2 = 10 & go to step 8.

Step 5: Read den.

Step 6: Put r2.value(den) & go to step 10.

Step 7: Put r2.print () & go to step 11.

Step 8: Use ratio (int a) & declare a=a1 then
~~ratio = (double) a/b & then return.~~

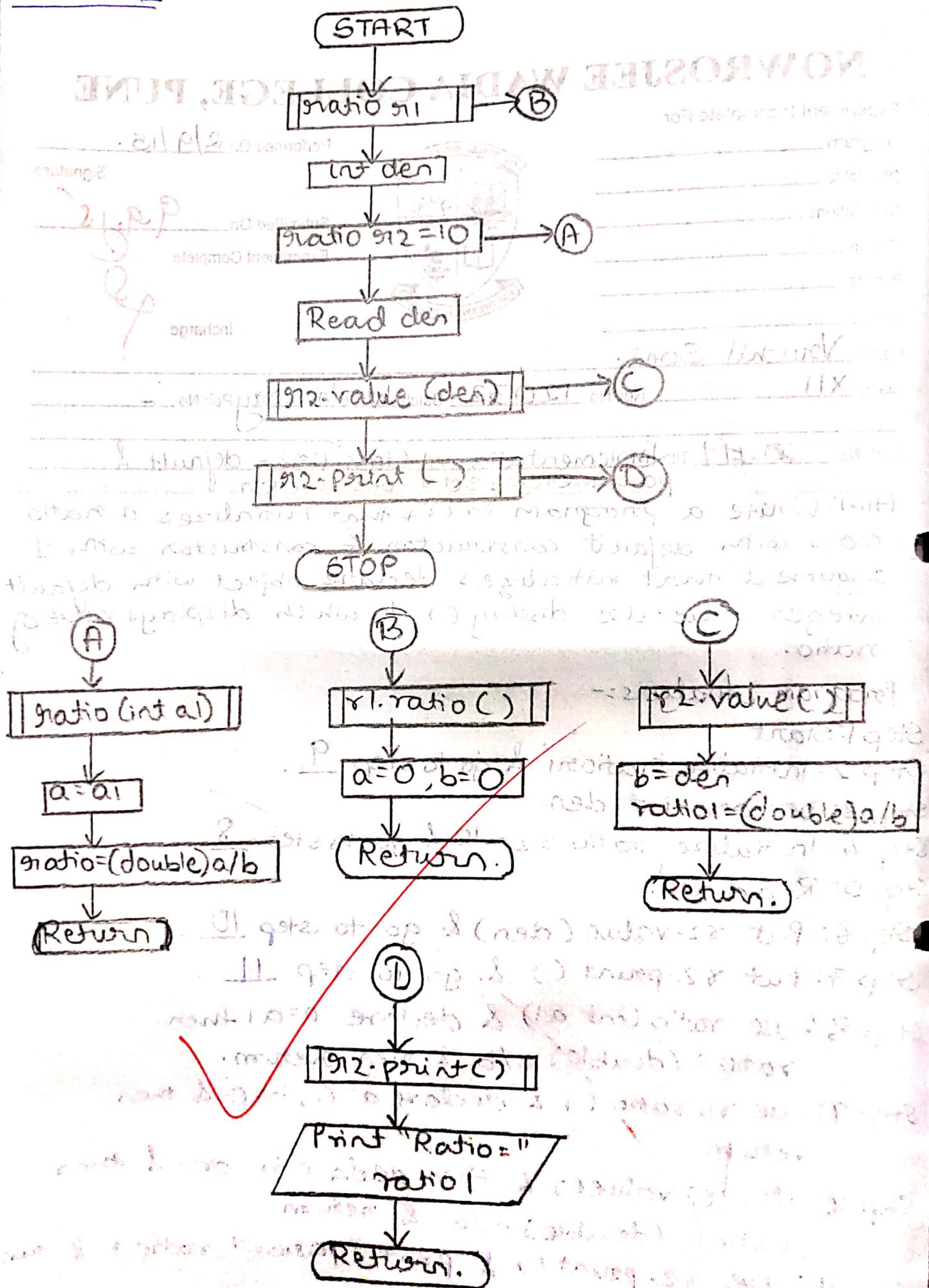
Step 9: Use r1.ratio () & declare a=0, b=0 & then
~~return.~~

Step 10: Use ~~r2.value()~~ & then declare b=den & then
~~ratio1 = (double) a/b & return.~~

Step 11: Use r2.print () & print "ratio=" ratio1 & then
~~return.~~

Step 12: STOP.

Flowchart:-



PARAMETE.CPP

```
//program performed by 12032
//program to implement class using default and
parameterised constructor
#include<iostream.h>
#include<conio.h>
class ratio
{
    int a,b;
    double ratio;
public:
    ratio()//default constructor
    {
        a=0;
        b=0;
    }
    ratio(int a1)//parameterised constructor
    {
        a=a1;
        ratio=(double)a/b;
    }
    void print()
    {
        cout<<"numerator is"<<a<<endl;
        cout<<"denominator is"<<b<<endl;
        cout<<"ratio is"<<ratio<<endl;
    }
    void value(int b1)
    {
        b=b1;
        ratio=(double)a/b;
    }
};
void main()
{
    clrscr();
    int den;
    ratio r1;
    ratio r2=10;//invokes parameterised constructor
}
```

PARAMETE.CPP

```
cout<<"enter denominator"<<endl;
```

```
cin>>den;
```

```
r2.value(den);
```

```
r2.print();
```

```
getch();
```

```
/*enter denominator
```

```
5
```

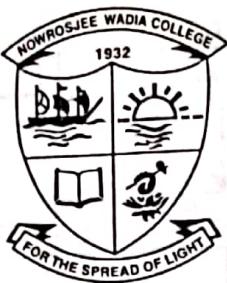
numerator is10

denominator is5

ratio is2*/

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____
Obs. Table _____
Calculations _____
Graphs _____
Results _____
Unit _____



Performed On 2/9/15

Signature

Submitted On 9.9.15

Experiment Complete

Incharge

Name Vrushil Soni
Class XII Roll No. 12032 Batch Wednesday Pair No. _____

Expt. No. - 10 Title Implementation of Constructor & Destructor.

Aim: Write a program in C++ that initializes ratio class with no parameters as default constructor. Program must print message "Object is born" during initialization. It should display message "Object is alive" when 1st member function ratio x() is called. The program must display "Object dies" when class destructor is called for the object when it reaches to the end of its scope.

Program Analysis:-

Step 1: Start.

Step 2: Initialize ratio r1 & go to step 5

Step 3: Initialize r1. ratiox () & go to step 6.

Step 4: Initialize ~ratio () & go to step 7.

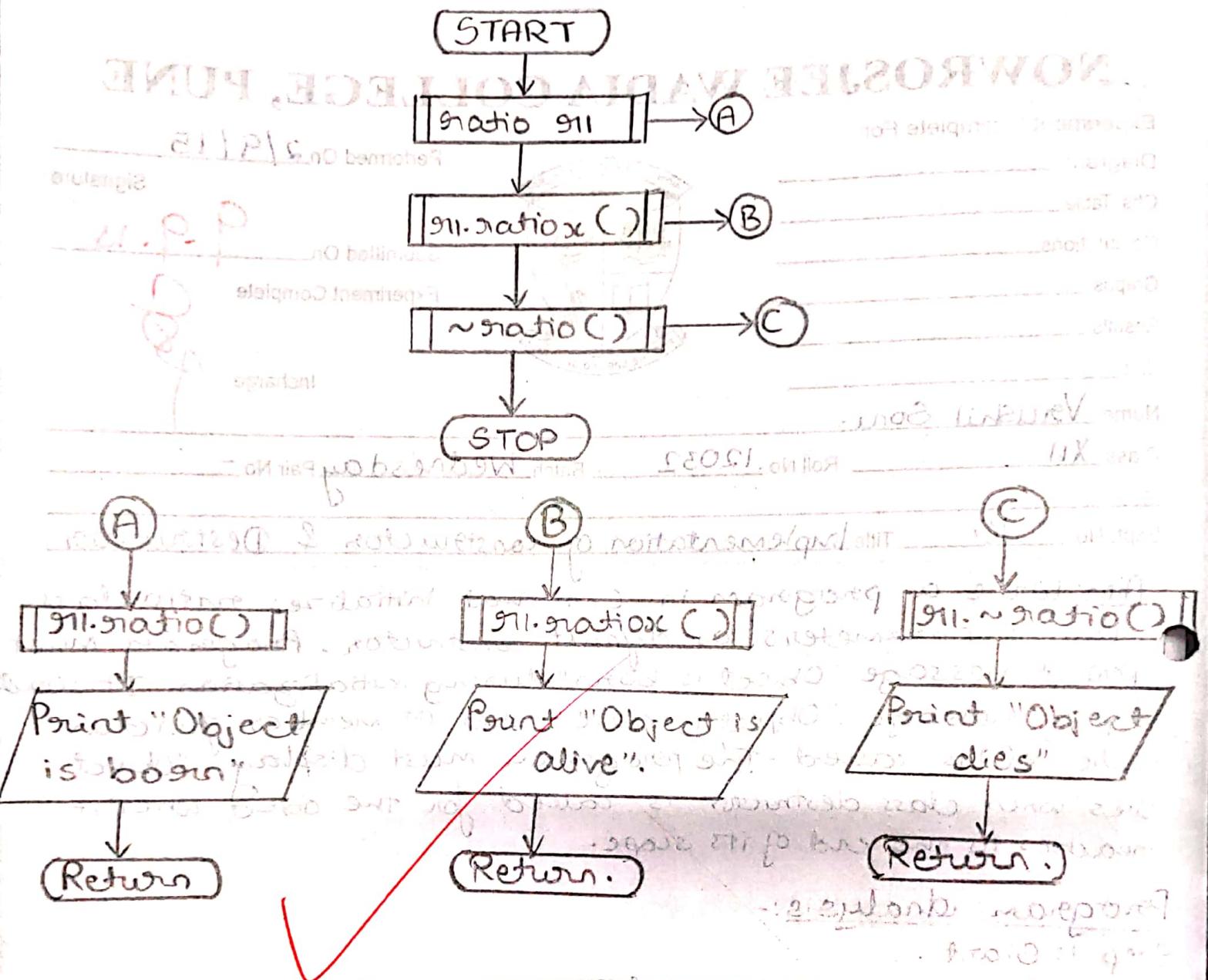
Step 5: Use function r1. ratio () & print "Object is born" & then return.

Step 6: Use r1. ratiox () & print "Object is alive" & then return.

Step 7: Use r1. ~ratio () & print "Object dies" & then return.

Step 8: Stop.

Flowchart :-



CONSTRDE.CPP

```
//program performed by 12032
//program for implementation of constructor and
destructor
#include<iostream.h>
#include<conio.h>
class ratio
{
public:
ratio();
void ratiox();
~ratio();
};
ratio::ratio()
{
cout<<"object is born"<<endl;
}
void ratio::ratiox()
{
cout<<"object is alive"<<endl;
}
ratio::~ratio()
{
cout<<"object dies"<<endl;
getch();
}
void main()
{
clrscr();
ratio r;
r.ratiox();
}
/*object is born
object is alive
object dies*/
```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
 Diagram _____
 Obs. Table _____
 Calculations _____
 Graphs _____
 Results _____
 Unit _____



Performed On 9/9/15 _____ Signature _____
 Submitted On 16.9.15 _____
 Experiment Complete _____

Name Vanshil Soni.

Class XII Roll No. 12032 Batch Wednesday Pair No. -

C
G

Expt. No. 11 Title Implementation of Class Complex.

Aim: Write a program in C++ with complex constructor even 2 complex no. i.e. $a = 4+5i$ & $b = 2+7i$. Program should print even a complex no. & their sum using user defined function sum().

Program analysis:-

Step 1: Start.

Step 2: Initialize complex c1 & go to step 10.

Step 3: Initialize complex c2 (4,5) & go to step 11.

Step 4: Initialize complex c3 (2,7) & go to step 11.

Step 5: Initialize c4 & go to step 10.

Step 6: Calculate $c4 = c2 \cdot \text{sum}(c3)$ & go to step 12.

Step 7: Display the fn using $c2 \cdot \text{display}()$ & go to step 13.

Step 8: Display the fn using $c3 \cdot \text{display}()$ & go to step 13.

Step 9: Display the fn using $c4 \cdot \text{display}()$ & go to step 14.

Step 10: Use fn complex() & declare $g1=0, i=0$ & return.

Step 11: Use fn complex (int a, int b) & declare $g1=a1, i=b1$ & return.

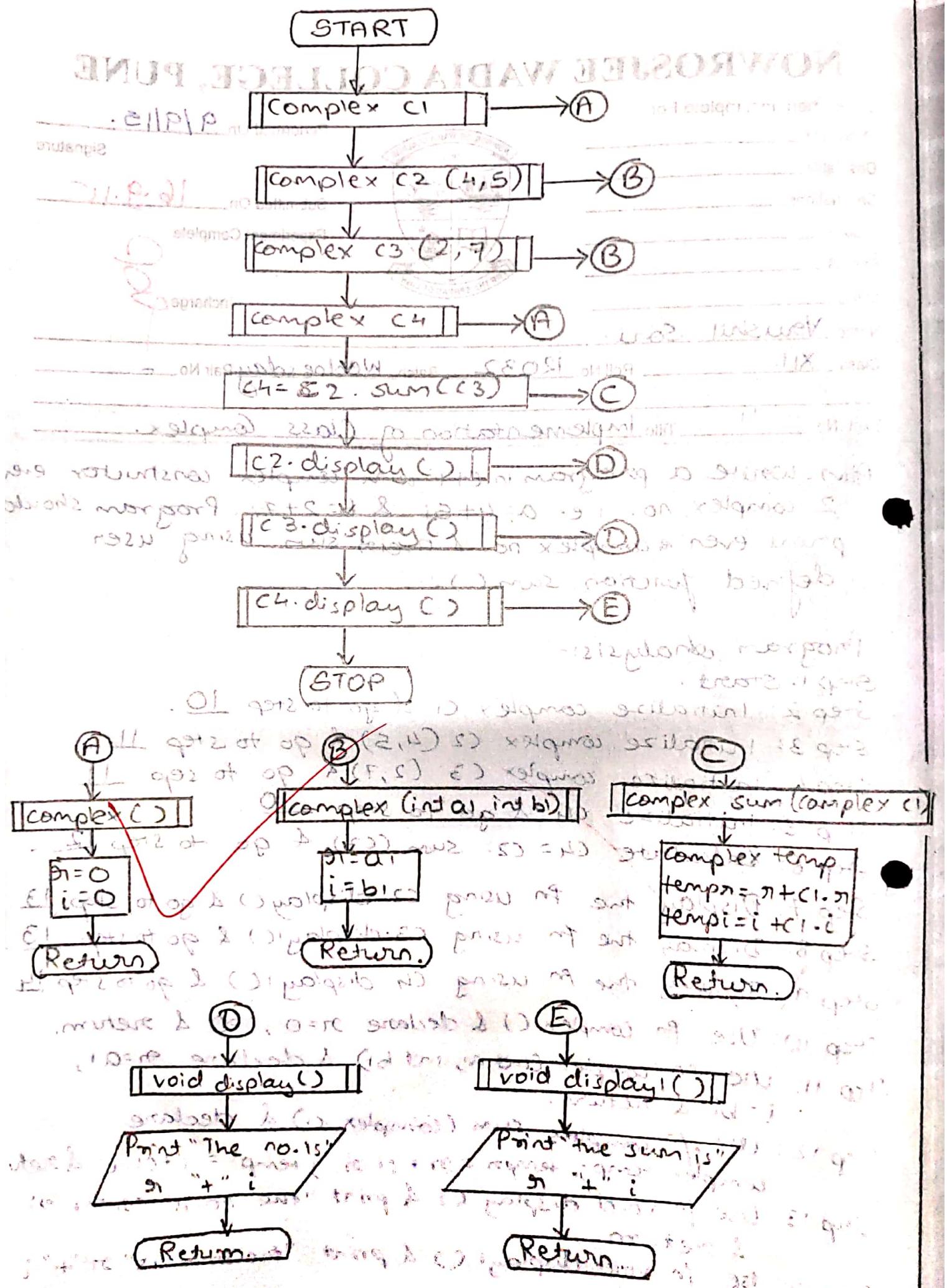
Step 12: Use fn complex sum (complex c1) & declare complex temp; $\text{temp}_r = g1 + c1.g1$; $\text{temp}_i = i + c1.i$ & return.

Step 13: Use fn void display() & print "the no. is" $g1+i$ & return.

Step 14: Use fn void display() & print "the sum is" $g1+i$ & return.

Step 15: Stop.

Flowchart:-



CLSCOMPL.CPP

```
//program performed by 12032
//program to implement class complex
#include<iostream.h>
#include<conio.h>
class complex
{
    int r,i,sum1,sum2;
public:
    complex()
    {
        r=0;
        i=0;
    }
    complex(int a1,int b1)
    {
        r=a1;
        i=b1;
    }
    void sum(complex c1,complex c2)
    {
        sum1=c1.r+c2.r;
        sum2=c1.i+c2.i;
        cout<<"the sum is "<<sum1<<"+ "<<sum2<<"i"<<endl;
    }
};
void main()
{
    clrscr();
    complex c1;
    complex c2(4,5);
    complex c3(2,7);
    complex c4;
    c4.sum(c2,c3);
    getch();
}/*the sum is 6+12i*/
```

```
//program performed by 12032 COMPLX2.CPP
//implementation of class complex method 2
#include<iostream.h>
#include<conio.h>
class complex
{
    int r,i;
public:
    complex()
    {
        r=0;
        i=0;
    }
    complex(int a1,int b1)
    {
        r=a1;
        i=b1;
    }
    complex sum(complex c1)
    {
        complex temp;
        temp.r=r+c1.r;
        temp.i=i+c1.i;
        return(temp);
    }
    void display()
    {
        cout<<"the no. is "<<r<<"+"<<i<<"i"<<endl;
    }
    void display1()
    {
        cout<<"the sum is "<<r<<"+"<<i<<"i"<<endl;
    }
};
void main()
{
    clrscr();
    complex c1;
    complex c2(4,5);
    complex c3(2,7);
    complex c4;
    c4=c2.sum(c3);c2.display();
    c3.display();
    c4.display1();
}
```

COMPLX2.CPP

```
getch();  
}/*the no. is 4+5i  
the no. is 2+7i  
the sum is 6+12i */
```

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____
Obs. Table _____
Calculations _____
Graphs _____
Results _____
Unit _____



Performed On 23/9/15

Signature

Submitted On 23/9/15 23.9.15

Experiment Complete

Incharge

Name Vanshil Soni

Class XII

Roll No. 12032

Batch Wednesday Pair No. -

Expt. No. 12 Title Implementation of '+' & '/' operator overloading

Aim: Write a program in C++ to implement '+' & '/' operator for ratio class using operator overloading concept.

Program Analysis:-

Step 1: Start.

Step 2: Use function ratio r & go to step 15.

Step 3: Use function ratio r1(4,5) & go to step 16.

Step 4: Initialize function ratio r2(3,7).

Step 5: Print "first ratio is"

Step 6: Use function r1.display() & go to step 17.

Step 7: Print "second ratio is"

Step 8: Use function r2.display() & go to step 17.

Step 9: Use function ratio r3=r1+r2 & go to step 19.

Step 10: Print "sum of ratio is"

Step 11: Use function r3.display() & go to step 17.

Step 12: Use function ratio r4=r1/r2 & go to step 21.

Step 13: Print "division of ratio is".

Step 14: Use function r4.display() & go to step 17.

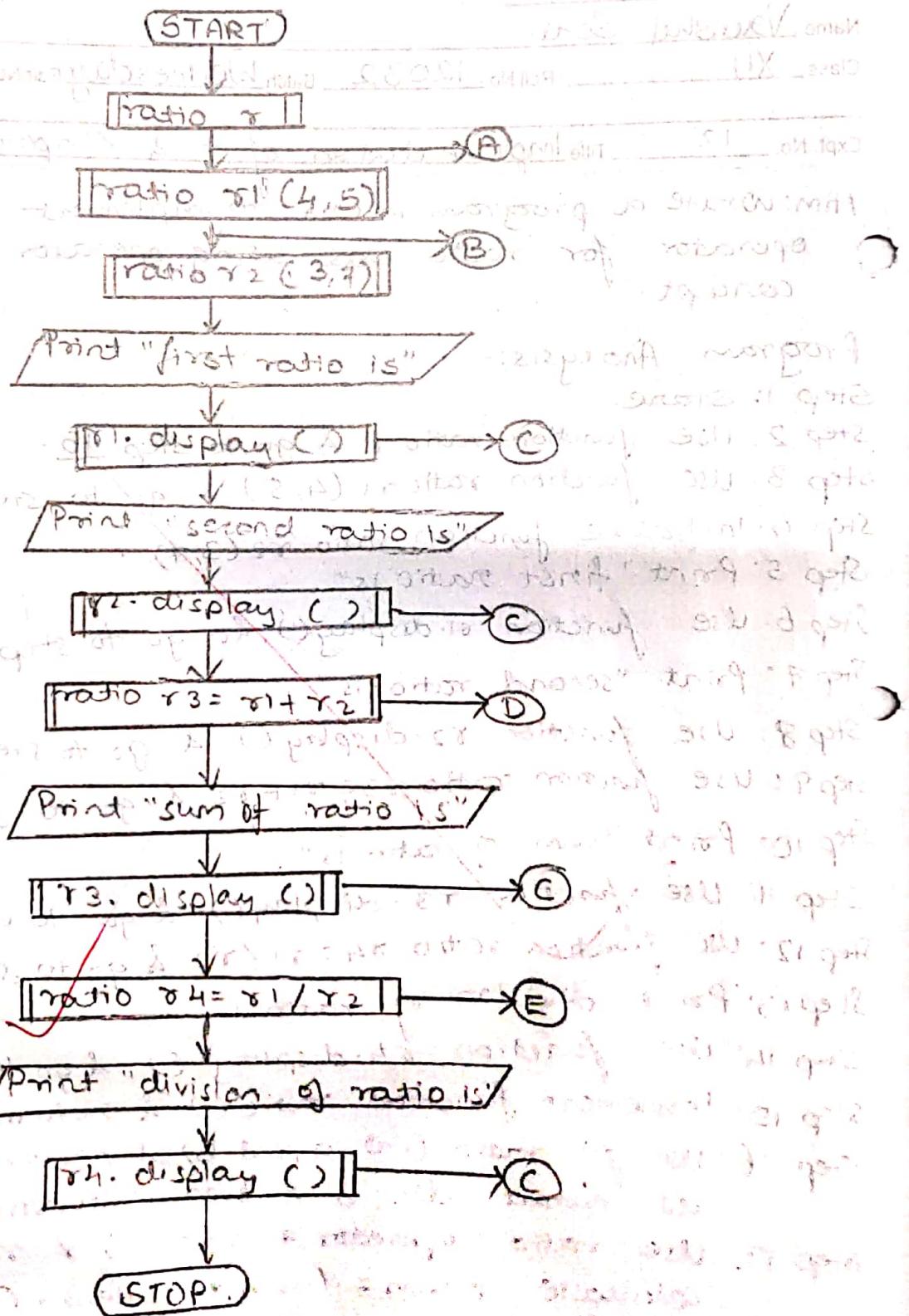
Step 15: Implement function ratio() & return.

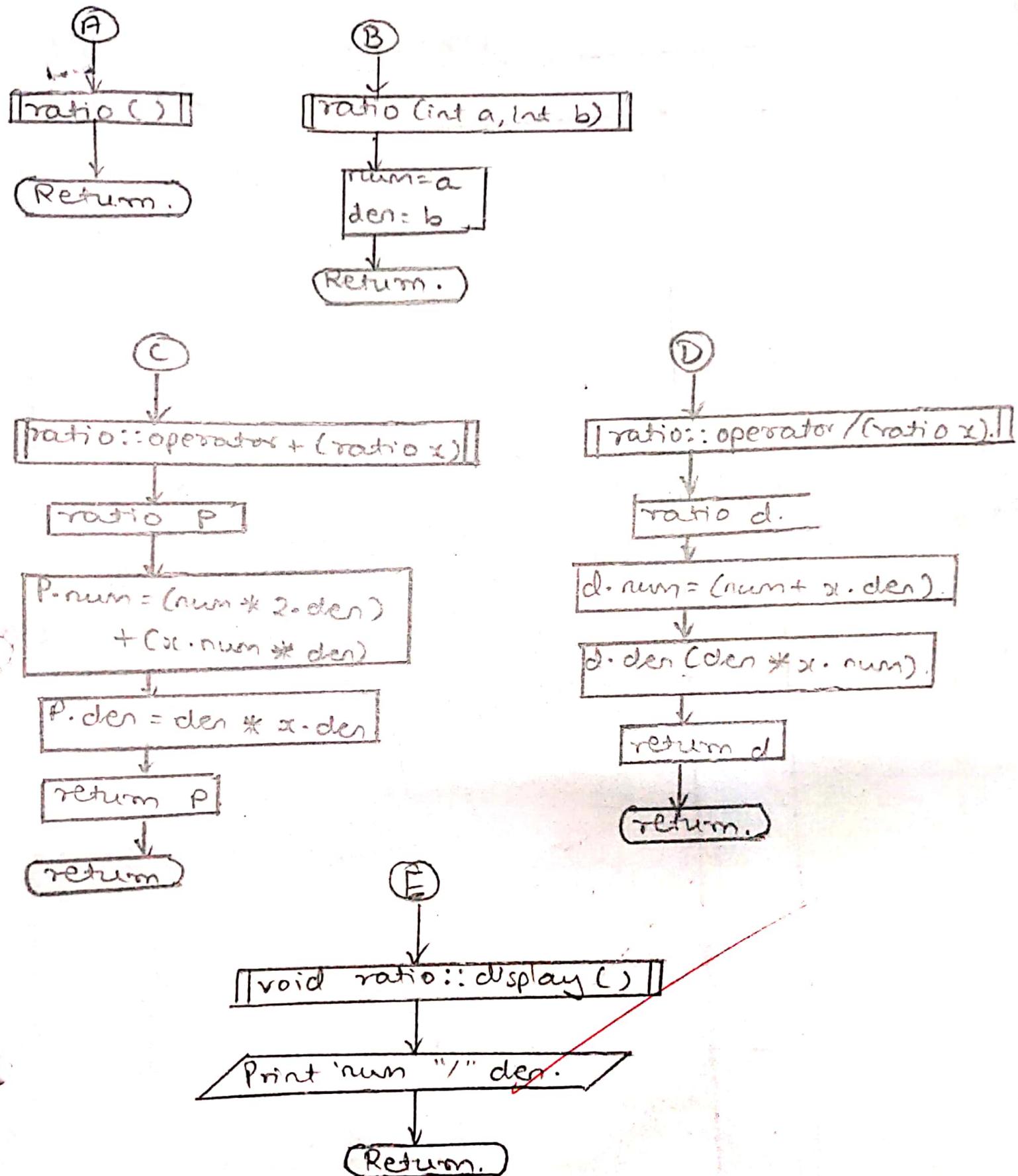
Step 16: Use f^n ratio (int a,int b) & declare variables as num=a, den=b & then return.

Step 17: Use ratio:: operator + (ratio x) & ratio p & calculate p.num = (num * 2.den) + (x.num * den).

- Step 18: Calculate $P \cdot den = den * x \cdot den$ & return P & return.
- Step 19: Use ratio::operator / (ratio x) & ratio d.
- Step 20: calculate $d \cdot num = (num + x \cdot den)$ & $d \cdot den (den * x \cdot num)$ & return d & then return.
- Step 21: Use function void ratio::display () .
- Step 22: Print num "/" den & then return.
- Step 23: Stop.

Flowchart:-





OVERLOAD.CPP

```
//program performed by 12032
//program to implement '+' and '/' operator
overloading
#include<iostream.h>
#include<conio.h>
class ratio
{
    int num, den;
public:
    ratio()
    {}
    ratio(int a,int b)
    {
        num=a;
        den=b;
    }
    ratio operator +(ratio);
    ratio operator /(ratio);
    void display();
};

ratio ratio::operator +(ratio x)
{
    ratio p;
    p.num=(num*x.den)+(x.num*den);
    p.den=den*x.den;
    return(p);
}
ratio ratio::operator /(ratio x)
{
    ratio d;
    d.num=(num*x.den);
    d.den=(den*x.num);
    return(d);
}
void ratio::display()
{
    cout<<num<<" / "<<den<<endl;
}
```

OVERLOAD.CPP

```
void main()
{
clrscr();
ratio r1(4,5);
ratio r2(3,7);
cout<<"first ratio is: "<<endl;
r1.display();
cout<<"second ratio is: "<<endl;
r2.display();
ratio r3=r1+r2;
cout<<"the sum of the ratio is: "<<endl;
r3.display();
ratio r4=r1/r2;
cout<<"the division of the ratio is: "<<endl;
r4.display();
getch();
}
```

/*first ratio is:
4/5

second ratio is:

3/7

the sum of the ratio is:

43/35

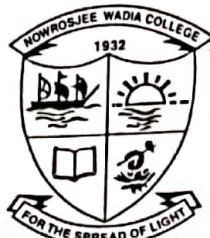
the division of the ratio is:

28/15*/

re3

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
 Diagram _____
 Obs. Table _____
 Calculations _____
 Graphs _____
 Results _____
 Unit _____



Performed On 30/9/15

Signature

Submitted On 14/10/15

Experiment Complete

G

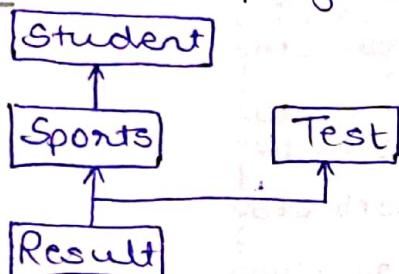
Incharge

Name Vrushil Joshi

Class XII Roll No. 12032 Batch. Wednesday Pair No. -

Expt. No. 13 Title Implementation of Hybrid inheritance.

Aim: Write a program in C++ to implement following diagram



Program Analysis:-

Step 1: Start.

Step 2: Use function result or

Step 3: Use member function r.getdata() & go to step 7.

Step 4: Use member fn r.sports_m() & go to step 10.

Step 5: Use member fn r.test_m() & go to step 12.

Step 6: Use function r.sum_sm_tm() & go to step 14.

Step 7: Use member fn r.showdata() & go to step 16.

Step 8: Use member fn r.display() & go to step 15.

Step 7: Initialize void getdata() & print "Enter name".

Step 8: Input the values using cin.getline (name, 30) & print "Enter roll no".

Step 9: Read as N & then return.

Step 10: Initialize void sports_m() & print "Enter sports marks".

Step 11: Read as sm and then return.

Step 12: Print "Enter sports t1 & t2 marks" & read as tm₁, tm₂ & then return.

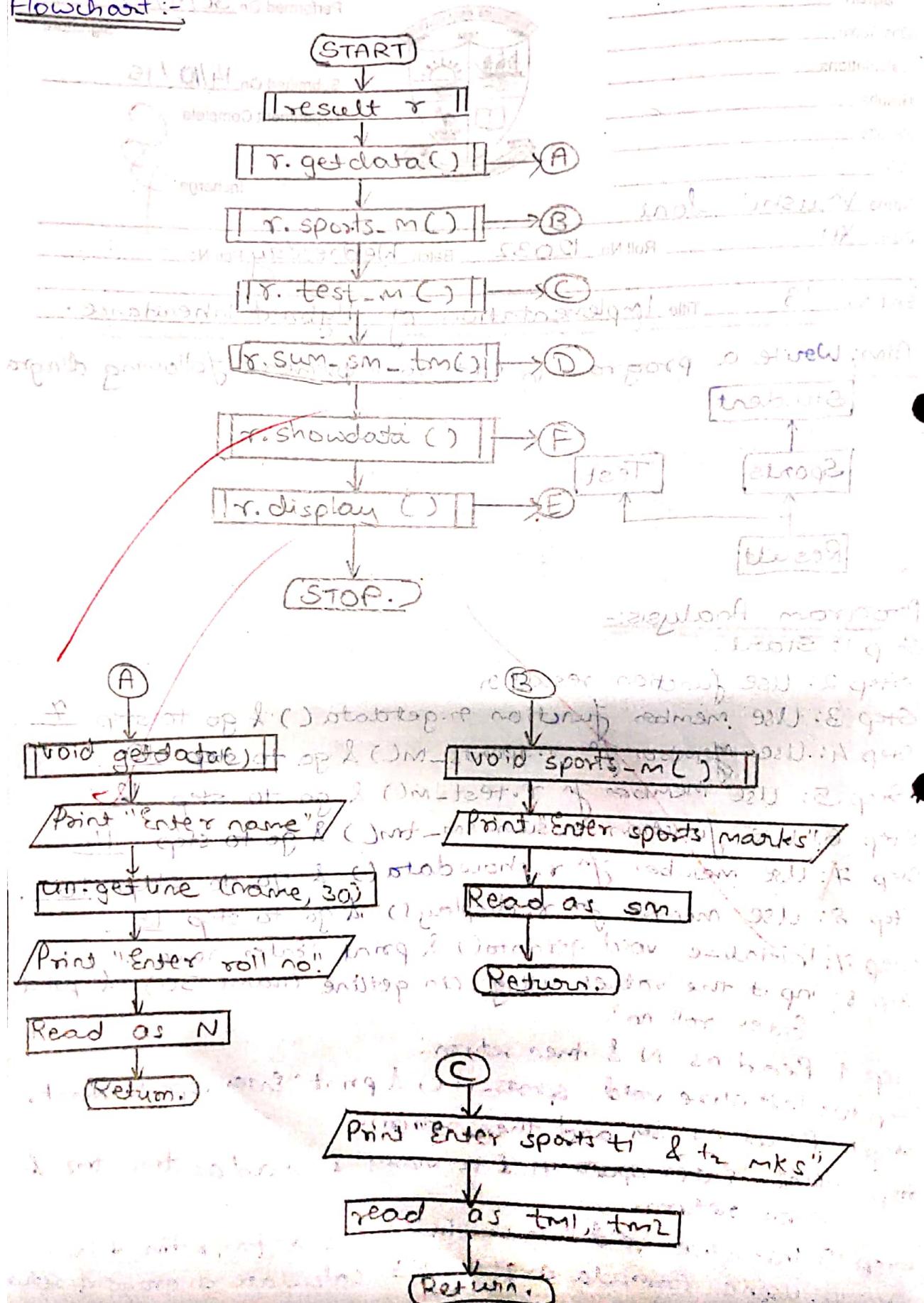
Step 13: Initialise void sum_sm_tm() & sum=tm₁+tm₂+sm.

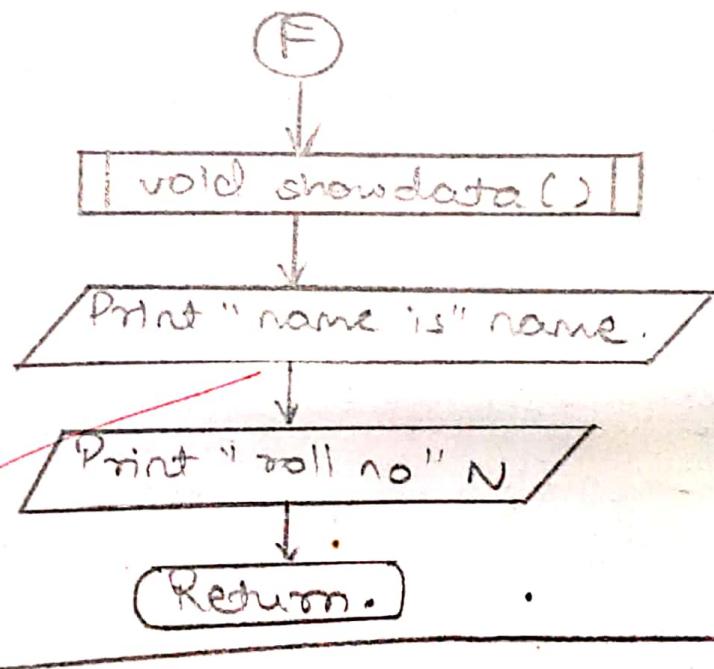
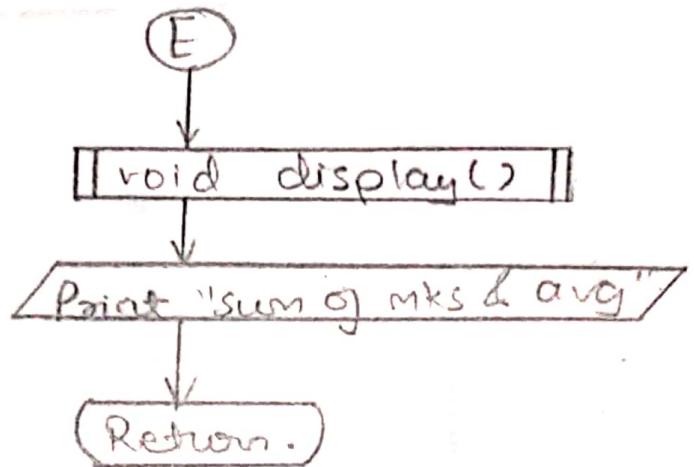
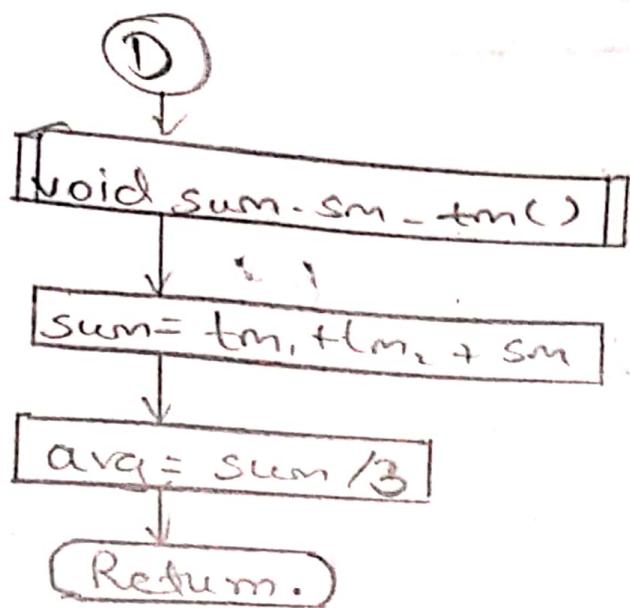
Step 14: Using formula avg= sum/3 calculate average & return.

Step 15: Use fn void display() & print "sum of mks & avg" & then return.

- Step 16: Use fn void showdata() & print "name is" name.
 Step 17: Print "roll no" & return.
 Step 18: Stop.

Flowchart:





HANDLING.CPP

```
//program performed by roll no 12032          HYBRIDEI.CPP
//program performed on implementation of hybride
inheritance
#include<iostream.h>
#include<conio.h>
#include<iomanip.h>
class student
{
protected:
int roll;
char name[20];
public:
void getdata()
{
cout<<endl<<setw(21)<<"enter name";
cin.getline(name,20);
cout<<endl<<setw(28)<<"enere roll no. ";
cin>>roll;
}
void showdata()
{
cout<<endl<<endl<<setw(18)<<"out put is "<<endl<<endl;
cout<<setw(15)<<"name: "<<name<<endl<<endl;
cout<<setw(19)<<"roll no.: "<<roll<<endl<<endl;
};
class sports:public student
{
protected:
float sm;
public:
void sports_m()
{
cout<<endl<<setw(33)<<"enter the sports marks: ";
cin>>sm;
}
};
class test
{
protected:
float tm1,tm2;
public:
void test_m()
{
```

```

HYBRIDEI.CPP
cout<<endl<<setw(31)<<"enter 1st test marks: ";
cin>>tm1;
cout<<endl<<setw(31)<<"enter 2nd test marks: ";
cin>>tm2;
};

class result:public sports, public test
{
protected:
float sum,avg;
public:
void sum_sm_tm()
{
sum=tm1+tm2+sm;
avg=sum/3;
}
void display()
{
cout<<setw(23)<<"sports marks: "<<sm<<endl<<endl;
cout<<setw(25)<<"1st test marks: "<<tm1<<endl<<endl;
cout<<setw(25)<<"2st test marks: "<<tm2<<endl<<endl;
cout<<setw(24)<<"total marks of "<<name<<"are:
"<<sum<<endl<<endl;
cout<<setw(24)<<"the avg of "<<name<<"is:
"<<avg<<endl<<endl;
}
};
void main()
{
clrscr();
result a;
a.getdata();
a.sports_m();
a.test_m();
a.sum_sm_tm();
a.showdata();
a.display();
getch();
}

/*          enter name vrushil

```

HYBRIDEI.CPP
enere roll no. 12032

enter the sports marks: 10

enter 1st test marks: 9

enter 2nd test marks: 9

out put is

name: vrushil

roll no.: 12032

sports marks: 10

1st test marks: 9

2st test marks: 9

total marks of vrushil is: 28

the avg of vrushil is: 9.333333
*/

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____

Obs. Table _____

Calculations _____

Graphs _____

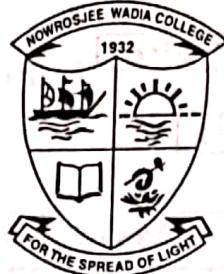
Results _____

Unit _____

Name Vrushil Soni.

Class XII

Roll No. 12032 Batch Wednesday Pair No. _____



Performed On 14/10/15

Signature _____

Submitted On 21/10/15

Experiment Complete

Incharge G

Expt. No. 14 Title Implementation of Virtual function

Aim: Write a program in C++ which implements the use of Virtual function.

Program Analysis:-

Step 1: Start.

Step 2: Declare person p1 & derive d1.

Step 3: ~~Declare~~ Initialize person * ptr.

Step 4: Declare ptr = & d1.

Step 5: Print "Derived class functions".

Step 6: Use function ptr → get() & go to step 12.

Step 7: Use function ptr → show() & go to step 15.

Step 8: Use ptr = & p1.

Step 9: Print "base class function".

Step 10: Use function ptr → get() & go to step 17.

Step 11: Use function ptr → show() & go to step 20.

Step 12: Print "Enter name" & then read name.

Step 13: Print "Enter address" & then read address.

Step 14: Print "Enter age" & then read age & return.

Step 15: Print "name:" (name)

Step 16: Print "address:" (address), "age:" (age) & return.

Step 17: Print "Enter name" & read name.

Step 18: Print "enter address" & then read address.

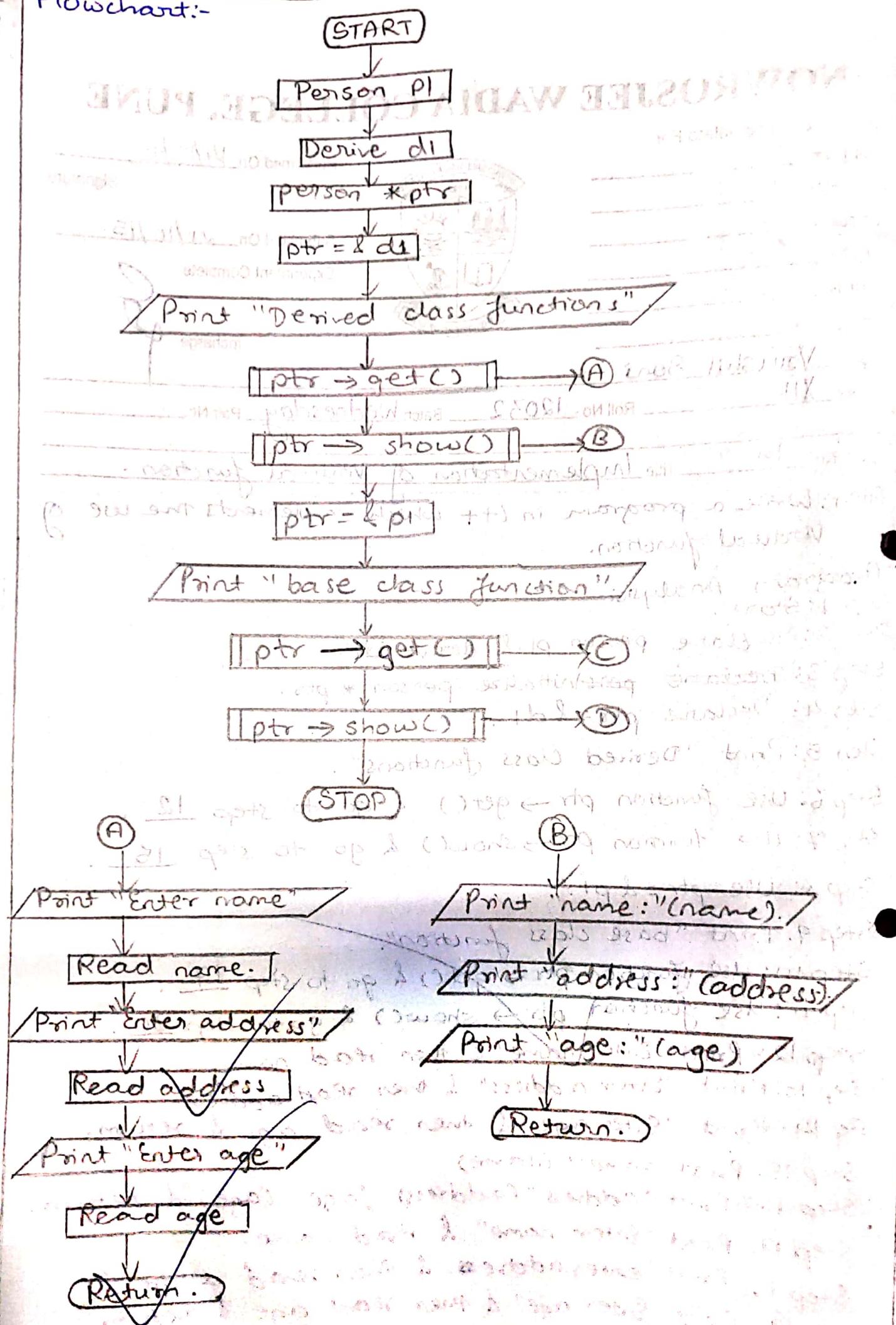
Step 19: Print "Enter age" & then read age & return.

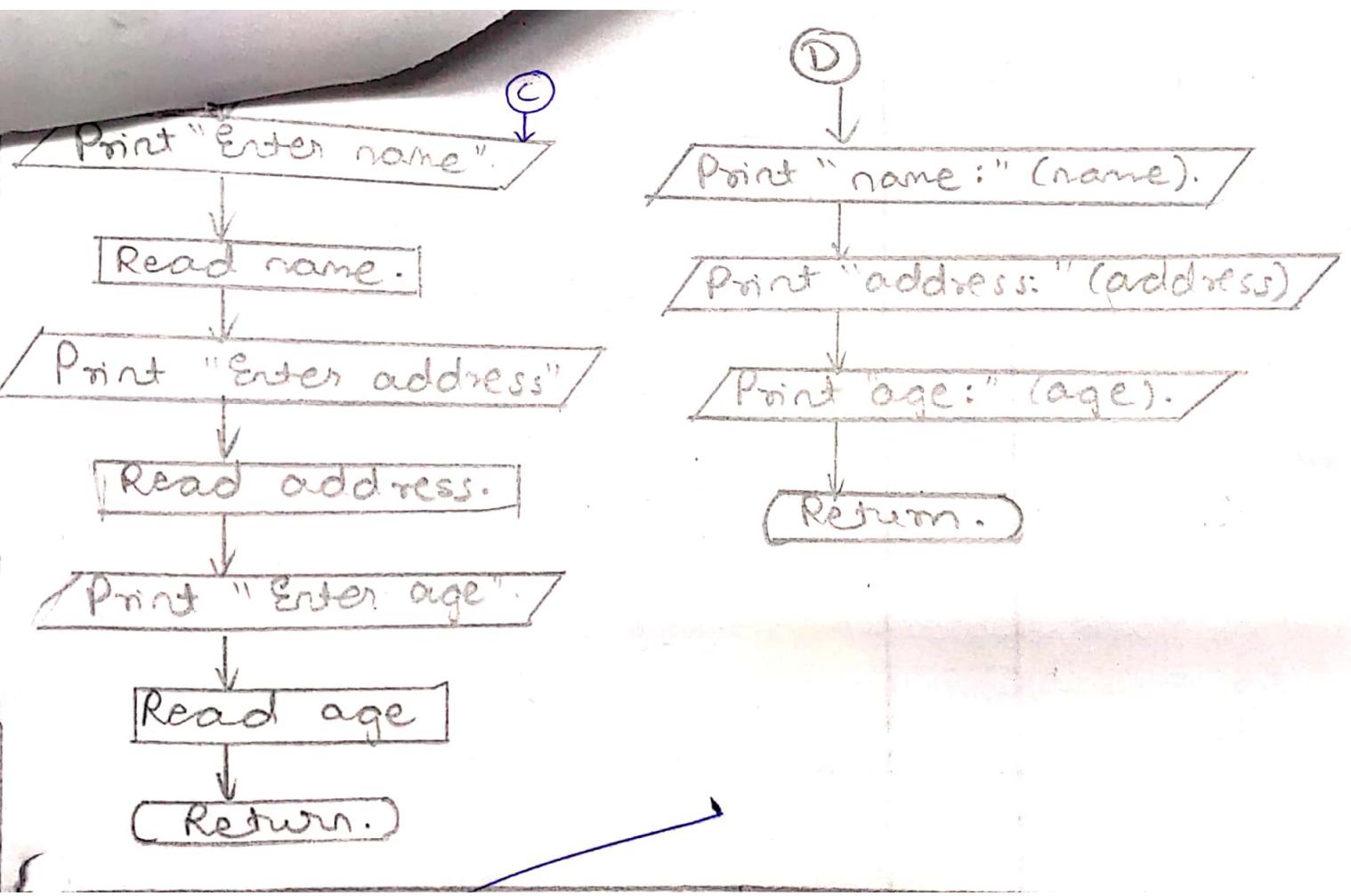
Step 20: Print "name:" (name)

Step 21: Print "address=" (address), "age:" (age) & then return.

Step 22: Stop.

Flowchart:-





VIRTUAL.CPP

```
//program performed by 12032
//program to implement virtual function
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
#include<string.h>
class person
{
protected:
int age;
char name[80], address[80];
public:
void virtual get()
{
cout<<endl<<"enter name:" ;
gets(name);
cout<<endl<<"enter address : ";
gets(address);
cout<<endl<<"enter age: ";
cin>>age;
}
void virtual show()
{
cout<<endl<<endl<<"Name: "<<name<<endl<<endl;
cout<<"address: "<<address<<endl<<endl;
cout<<"age: "<<age<<endl<<endl;
};
class derive:public person
{
protected:
int age;
char name[80], address[80];
public:
void get()
{
cout<<endl<<"enter name: ";
gets(name);
```

```

cout<<endl<<"enter address: ";
gets(address);
cout<<endl<<"enter age: ";
cin>>age;
}
void show()
{
cout<<endl<<endl<<"name: "<<name<<endl<<endl;
cout<<"address: "<<address<<endl<<endl;
cout<<"age: "<<age<<endl<<endl;
}
};

void main()
{
clrscr();
person p1;
derive d1;
person *ptr;
ptr=&d1;
cout<<"derived class functions: "<<endl;
ptr->get();
ptr->show();
ptr=&p1;
cout<<endl<<endl<<endl<<"base class functions:
"<<endl;
ptr->get();
ptr->show();
getch();
} /*name: vrushil

```

address: balaji nagar

age: 16

base class functions:

enter name:vrushil

enter address :wadia

enter age: 45

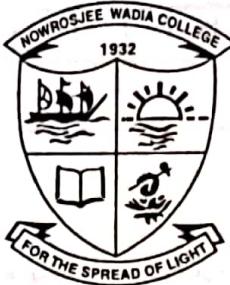
Name: vrushil

address: wadia

age: 45*/

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
Diagram _____
Obs. Table _____
Calculations _____
Graphs _____
Results _____
Unit _____



Performed On 21/10/15.

Signature

Submitted On 26/10/15 2.11.15

Experiment Complete

Incharge

Name Vrushil Soni

Class XII Roll No. 12032 Batch Wednesday Pair No. -

Expt. No. 15 Title File Handling.

Aim: write a program in c++ which reads country & capital & open that 2 files in the main program.

Program Analysis:-

Step 1: Start.

Step 2: Declare const int size=80. & char line [size].

Step 3: Declare ifstream fin1, fin2.

Step 4: Use fin1.open ("d:\country.txt") &
fin2.open ("d:\capital.txt")

Step 5: Create a for loop where i=1, i<6, i++.
If it is true then check if fin1.eof() == 0. If false go to step 9.

Step 6: If true then print "Exit from country & use exit(1) &

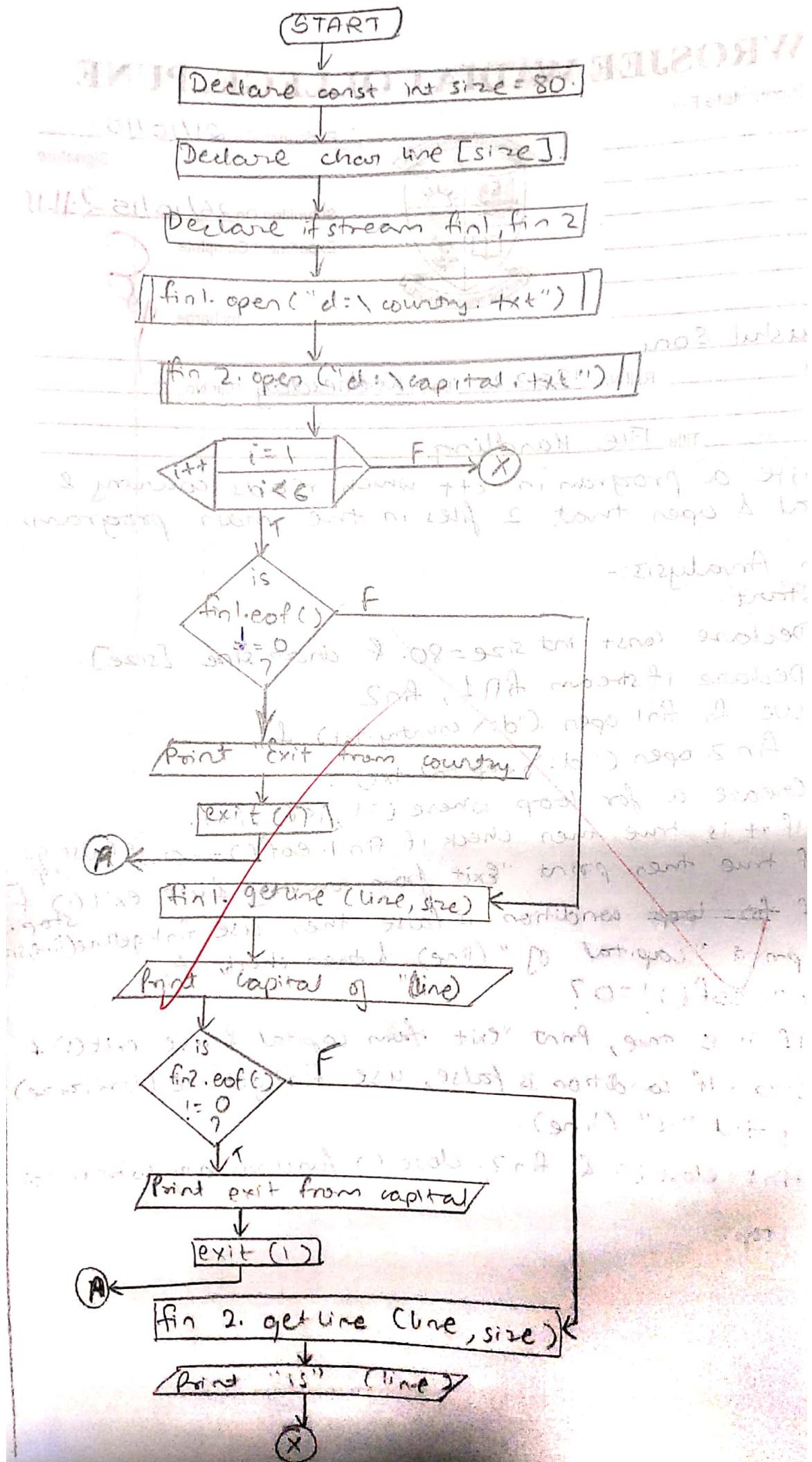
Step 7: If ~~for loop~~ condition is false then use fin1.getline(line, size)
& print "capital of " (line) & then check if
fin2.eof() != 0?

Step 8: If it is true, Print "exit from capital & use exit(1) &
Step. If condition is false, use fin2.getline (line, size)
& print "is" (line).

Step 9: fin1.close() & fin2.close () functions are to be used.

Step 10: Stop.

Flowchart:-





fin 1. close ()

fin 2. close ()



STOP

HANDLING.CPP

```
//program performed by 12032
//program for file handling
#include<iostream.h>
#include<conio.h>
#include<process.h>
void main()
{
    clrscr();
    const int size=80;
    char line[size];
    ifstream fin1, fin2;
    fin1.open("d:\country.txt");
    fin2.open("d:\capital.txt");
    for(int i=1;i<6;i++)
    {
        if(fin1.eof()!=0)
        {
            cout<<"exit from country";
            exit(1);
        }
        fin1.getline(line,size);
        cout<<"capital of "<<line;
        if(fin2.eof()!=0)
        {
            cout<<"exit from capital";
            exit(1);
        }
        fin2.getline(line,size);
        cout<<" is "<<line<<endl;
    }
    fin1.close();
    fin2.close();
    getch();
}
```

/*capital of India is Delhi
capital of USA is washington

capital of SRI LANKA is colombo

capital of NEPAL is kathmandu

capital of Bangladesh is Dhaka*/

NOWROSJEE WADIA COLLEGE, PUNE

Experiment Incomplete For
 Diagram _____
 Obs. Table _____
 Calculations _____
 Graphs _____
 Results _____
 Unit _____



Performed On 14/10/15

Signature

Submitted On 21/10/15

Experiment Complete

Incharge

Name Vrushil Joshi

Class XII Roll No. 12032 Batch Wednesday Pair No. -

Expt. No. 16 Title Conversion Routine

Aim: Define 2 classes polar & rectangular to represent points in the polar & rectangular system. Use conversion routines from 1 system to other.

Program Analysis:-

Step 1: Start.

Step 2: Declare classes rectangle rc1 and polar and go to Step 8.

Step 3: Declare class polar p1 and go to step 11.

Step 4: Use function con_rect(rc1) & go Print "rectangular to polar".

Step 5: Use function con_rect(rc1) & go to step 14.

Step 6: Print "Polar to rectangular".

Step 7: Use function con_polar(p1) & go to step 17.

Step 8: Print "Enter rectangular coordinates".

Step 9: Print "x=" & read x.

Step 10: Print "y=" & read y. & then return.

Step 11: Print "Enter polar coordinates"

Step 12: Print "r=" & read r.

Step 13: Print "theta=" & read t & then return.

Step 14: Print precision(2)

Step 15: Print "r=" (sqrt(a.x*a.x+a.y*a.y))

Step 16: Print "theta=" (atan(a.y/a.x)*(180/3.14)) & return.

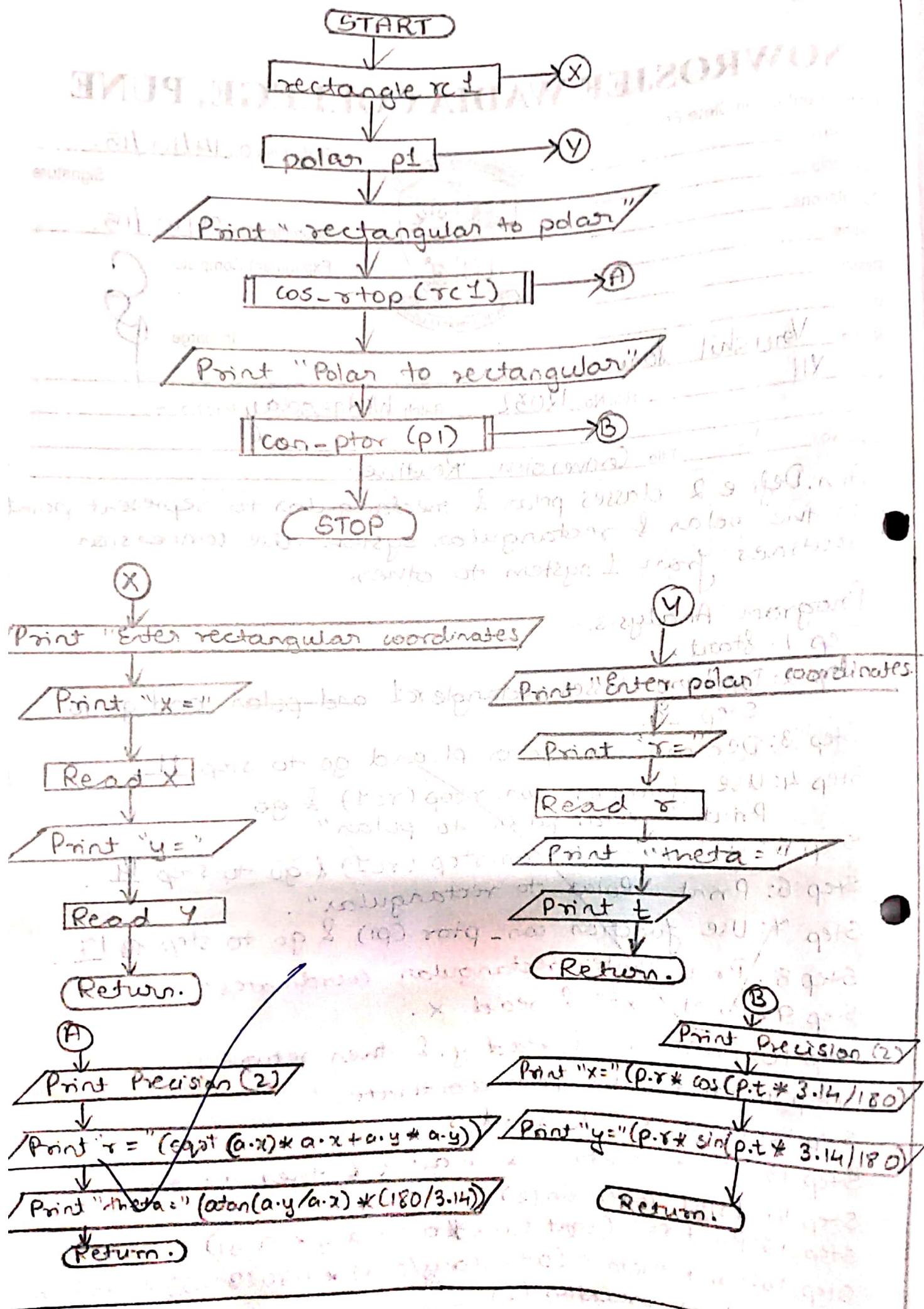
Step 17: Print precision(2)

Step 18: Print "x=" (p.r*cos(p.t*3.14/180))

Step 19: Print "y=" (p.r*sin(p.t*3.14/180)) & return.

Step 20: Stop.

Flowchart:-



```

//program to implement conversion routine           CONVERSI.CPP
//program performed by 12032
#include<iostream.h>
#include<conio.h>
#include<math.h>
#include<iomanip.h>
class rectangle
{
    float x,y;
public:
rectangle()
{
    cout<<"enter rectangle co-ordinates" <<endl <<endl;
    cout<<"x=" ;
    cin>>x;
    cout<<endl <<"y=" ;
    cin>>y;
}
friend void con_rtop(rectangle);
};
class polar
{
float r,t;
public:
polar()
{
    cout<<endl <<"enter polar coordinates" <<endl <<endl;
    cout<<"r=" ;
    cin>>r;
    cout<<endl <<"theta=" ;
    cin>>t;
}
friend void con_ptor(polar);
};
void con_rtop(rectangle a)//convert rectangle to polar
{
    cout.precision(2);
    cout<<"r=" <<sqrt(a.x*a.x+a.y*a.y) <<endl;
    cout<<"theta=" <<atan(a.y/a.x)*(180/3.14) <<endl;
}
void con_ptor(polar p)//convert polar to rectangular
{
    cout.precision(2);
    cout<<"x=" <<p.r*cos(p.t*3.14/180) <<endl;
}

```

```
CONVERSI.CPP
} cout<<"y ="<<p.r*sin(p.t*3.14/180)<<endl;
void main()
{
    clrscr();
    rectangle rc1;
    polar p1;
    cout<<endl<<"rectangle to polar"<<endl<<endl;
    con_rtop(rc1); //invokes friend from rectangle
    cout<<endl<<"polar to rectangular"<<endl<<endl;
    con_ptor(p1); //invokes friend from polar
    getch();
} /*enter rectangle co-ordinates
```

x=5

y=5

enter polar coordinates

r=5

theta=5

rectangle to polar

r=7.07
theta+45.02

polar to rectangular

x=4.98
y=0.44 */