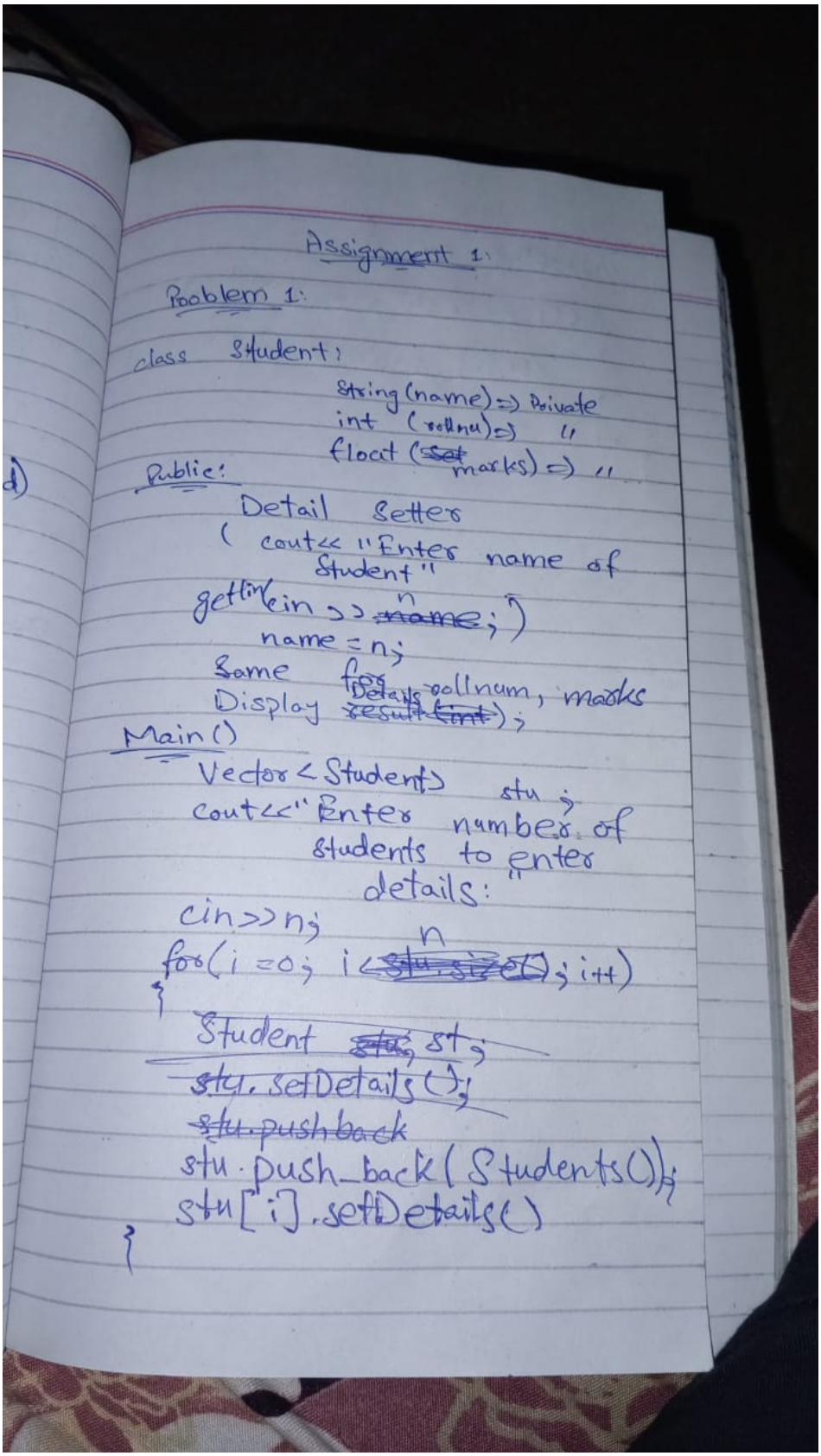
DSA Assignment 1 - (53457) M. Abdullah

Problem 1:

Create a simple **Student** class to store and manage the details of multiple students using an array of objects. **Class Student**:

- Data Members:
  - name (string)
  - rollNumber (int)
  - marks (float)
- Member Functions:
  - setDetails(): Function to input the student details
- Use an array of Student objects to store the details of multiple students.
- Allow the user to enter the details of n students and display their details.

Problem Solving



Assignment 1:

Problem 1:

class Student:

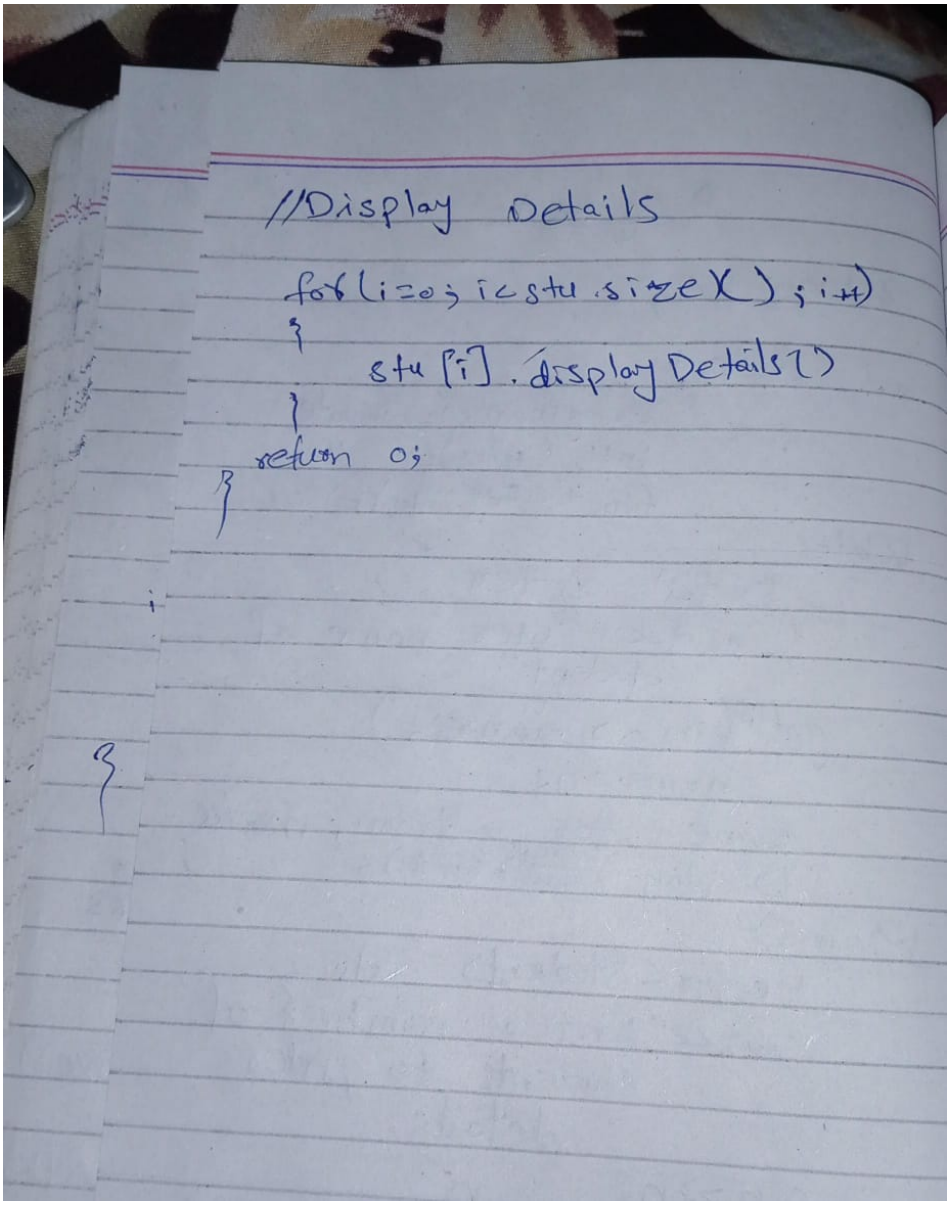
String (name) => Private
int (rollnu) => "
float (~~set~~ marks) => "

Public:

Detail Setter
(cout << "Enter name of Student"
getline(cin >> ~~name~~ n;)
name = n;
Same for rollnum, marks
Display ~~result~~ Details ~~int~~);

Main()

Vector < Student > stu;
cout << "Enter number of students to enter details:"
cin >> n;
for(i = 0; i < n; i++)
{
Student ~~stu~~ st;
~~stu.setDetails();~~
~~stu.push back~~
stu.push_back(Students());
stu[i].setDetails()
}



Code

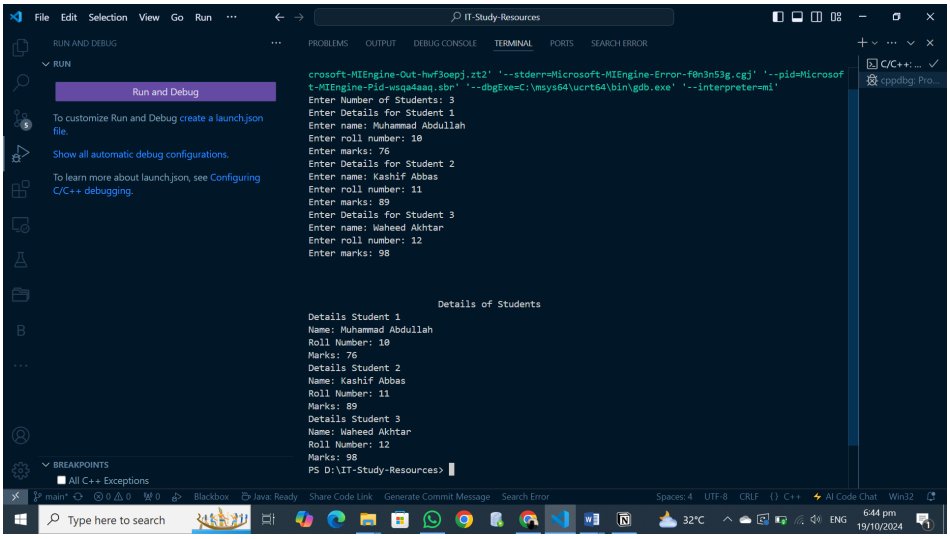
```
#include<iostream>
#include<string>
#include<vector>
using namespace std;

class Student {
    private:
```

```
        string name;
        int rollNum;
        float marks;
    public:
        void setDetails() {
            string na;int ro;float ma;
            cout<<"Enter name: ";
            cin.ignore();
            getline(cin,na);
            name =na;
            cout<<"Enter roll number: ";
            cin>>ro;
            rollNum = ro;
            cout<<"Enter marks: ";
            cin>>ma;
            marks = ma;
        }
        void displayDetails() {
            cout<<"Name: "<<name<<endl;
            cout<<"Roll Number: "<<rollNum<<endl;
            cout<<"Marks: "<<marks<<endl;
        }
};
int main() {
    int n;
    vector<Student> stu;
    cout<<"Enter Number of Students: ";
    cin>>n;
    for(int i=0;i<n;i++) {
        cout<<"Enter Details for Student "<<i+1<<endl;
        stu.push_back(Student());
        stu[i].setDetails();
    }
    cout<<"\n\n\n\t\t\tDetails of Students\t\t\t"<<endl;
    for(int i=0;i<stu.size();i++) {
        cout<<"Details Student "<<i+1<<endl;
```

```
            stu[i].displayDetails();
        }
        return 0;
    }
```

Output



Problem 2:

Create a simple program to calculate the area of different shapes using **inheritance**. You will create a base class for a general shape, and derive classes for specific shapes like **Circle** and **Rectangle**.

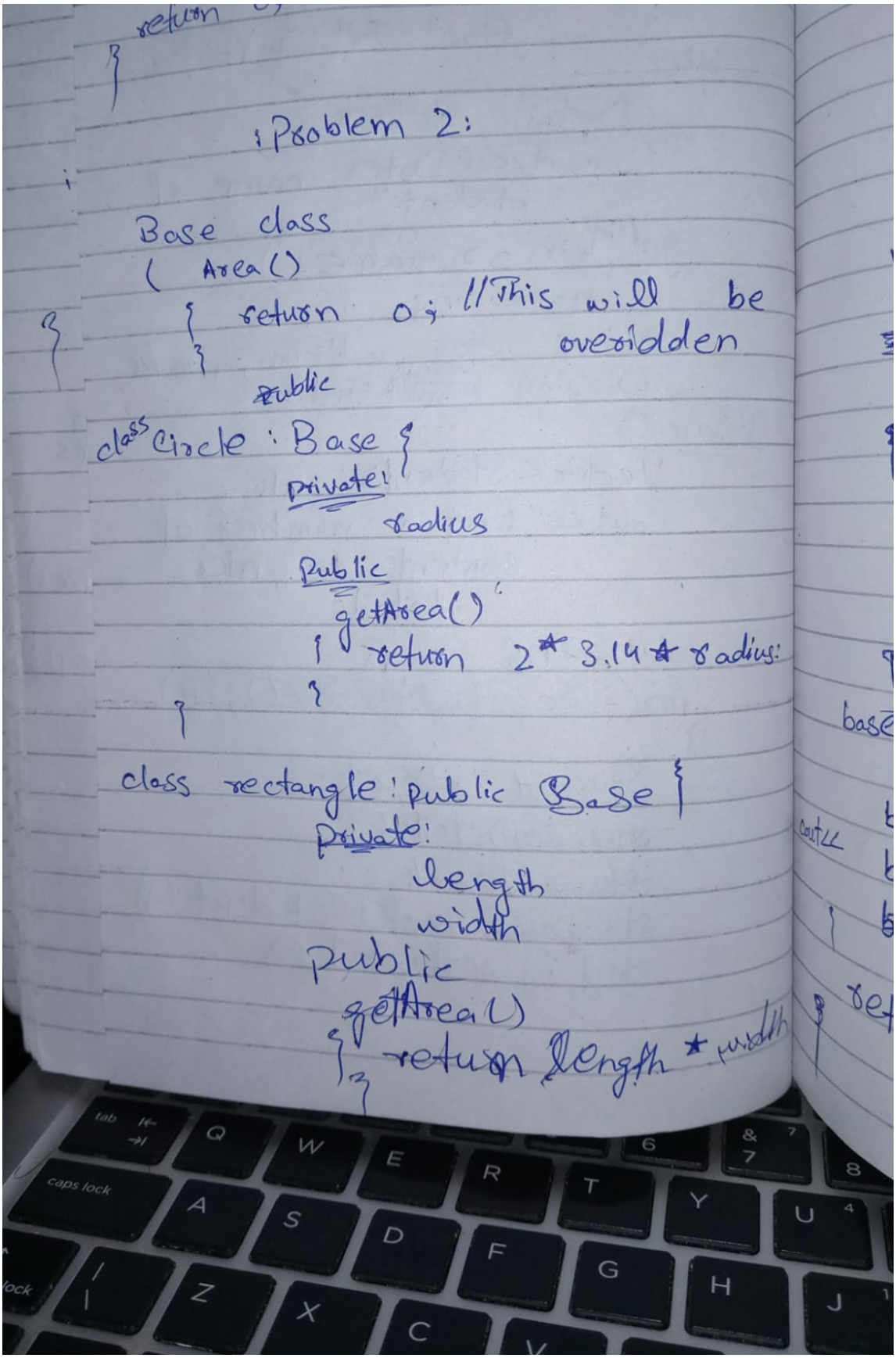
Requirements:

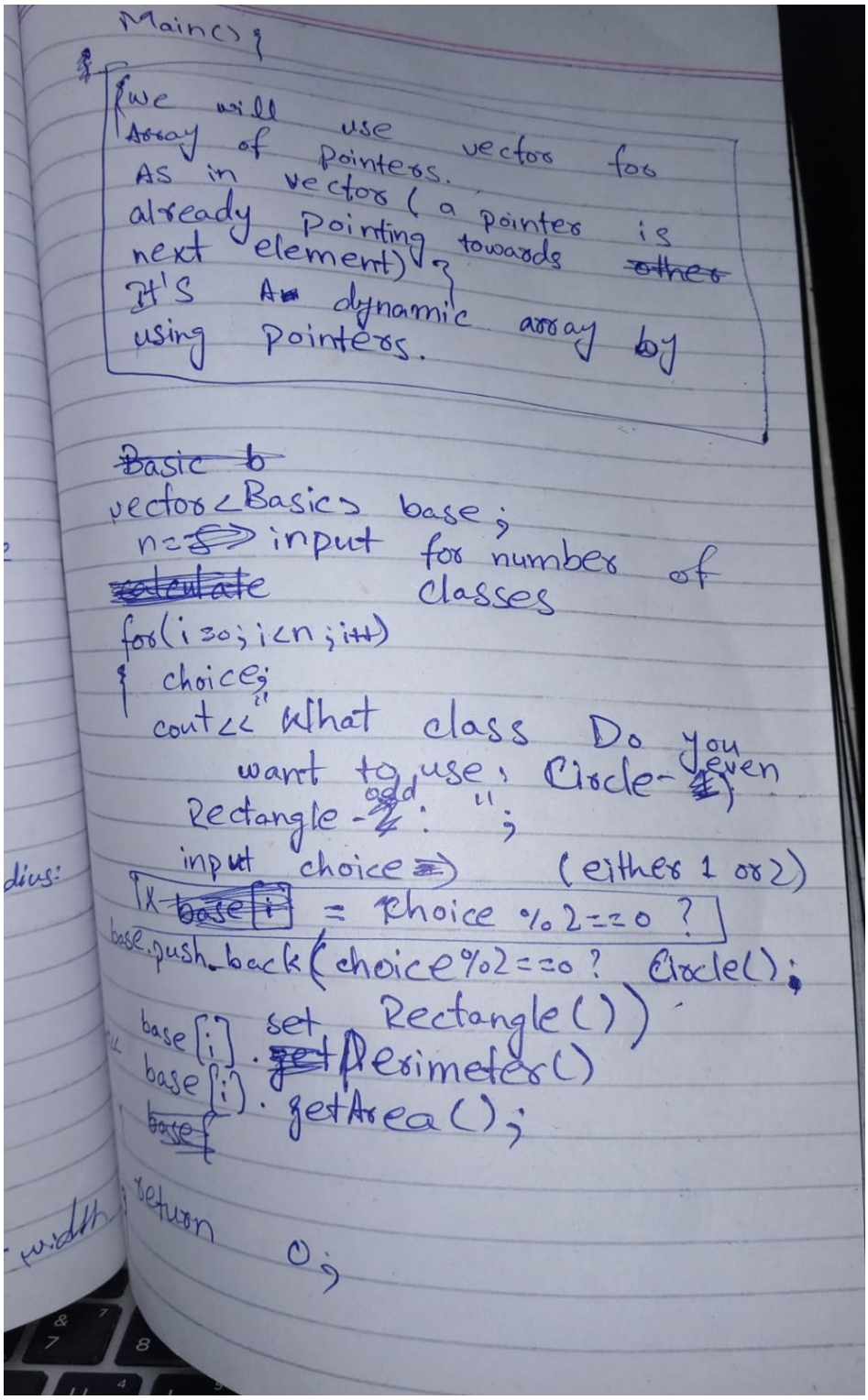
1. **Base Class Shape**:
    - Member Function:
        - getArea(): Pure virtual function to calculate the area of a shape.
2. **Derived Class Circle** (inherits from Shape):
    - Data Member:
        - radius (float)
    - Override getArea() to calculate the area of a circle.

3. **Derived Class Rectangle** (inherits from Shape):
    - Data Members:
        - length (float)
        - width (float)
    - Override getArea() to calculate the area of a rectangle.
4. **Main Task**:
    - Use an array of pointers to the base class to store different shapes.

Calculate and display the area of each shape.

Problem Solving


return
Problem 2:
Base class
Area()
return 0; //This will be overridden
public
class Circle : Base {
private:
radius
Public
getArea()
return 2 * 3.14 * radius;
class rectangle : public Base {
Private:
length
width
Public
getArea()
return length * width


Main() {
We will use vector for Array of Pointers.
As in vector (a pointer is already Pointing towards next element) ;
It's A dynamic array by using Pointers.
Basic b
vector<Basic> base;
n=> input for number of classes
for(i=0; i<n; i++)
{ choice;
cout<< "What class Do you want to use; Circle- even
Rectangle - odd : ";
input choice => (either 1 or 2)
base[i] = choice %2==0 ?
base.push_back(choice%2==0 ? Circle();
Rectangle())
base[i].set Perimeter()
base[i].getArea();
return 0;

Code

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

class Shape {
    double getArea() {
        return 0;
    }
};

class Circle: public Shape {
    private:
    double radius;
    public:
    void setPerimeter() {
        double rad;
        cout<<"Enter Radius for Circle: ";
        cin>>rad;
        radius = rad;
    }
    double getArea() {
    return 3.14*radius*radius;
    }
};

class Rectangle: public Shape {
    private:
    double length, width;
    public:
    void setPerimeter() {
        double len;
        cout<<"Enter Length for Rectangle: ";
        cin>>len;
        length = len;
```

```
        double wid;
        cout<<"Enter Width for Rectangle: ";
        cin>>wid;
        width = wid;
    }
    double getArea() {
        return (double) length*width;
    }
};

int main() {
    //vector<Shape> shapes;
    vector<Circle> circles;
    vector<Rectangle> rectangles;
    int n;
    cout<<"Enter number of shapes to find area of: ";
    cin>>n;
    for(int i = 0; i < n; i++) {
        int choice;
        cout<<"Enter even for Circle, odd for Rectangle: ";
        cin>>choice;
        //shapes.push_back(choice%2==0 ? Circle():Rectangle());
        if(choice%2==0) {
            circles.push_back(Circle());
            circles[i].setPerimeter();
            cout<<"Area of Circle "<<i+1<<": "<<circles[i].getAr
        }
        else {
            rectangles.push_back(Rectangle());
            rectangles[i].setPerimeter();
            cout<<"Area of Rectangle "<<i+1<<": "<<rectangles[i]
        }
    }
    /*cout<<"\n\t\t\t\tArea of Circles: \t\t\t\t\n";
    for(int i=0;i<circles.size();i++) {
        cout<<"Area of Circle "<<i+1<<": "<<circles[i].getArea(
```

```
    }
    cout<<"\n\t\t\t\tArea of Rectangles: \t\t\t\n";
    for(int i=0;i<rectangles.size();i++) {
        cout<<"Area of Rectangle "<<i+1<<": "<<rectangles[i].get
    }*/
    return 0;
}
```

Output