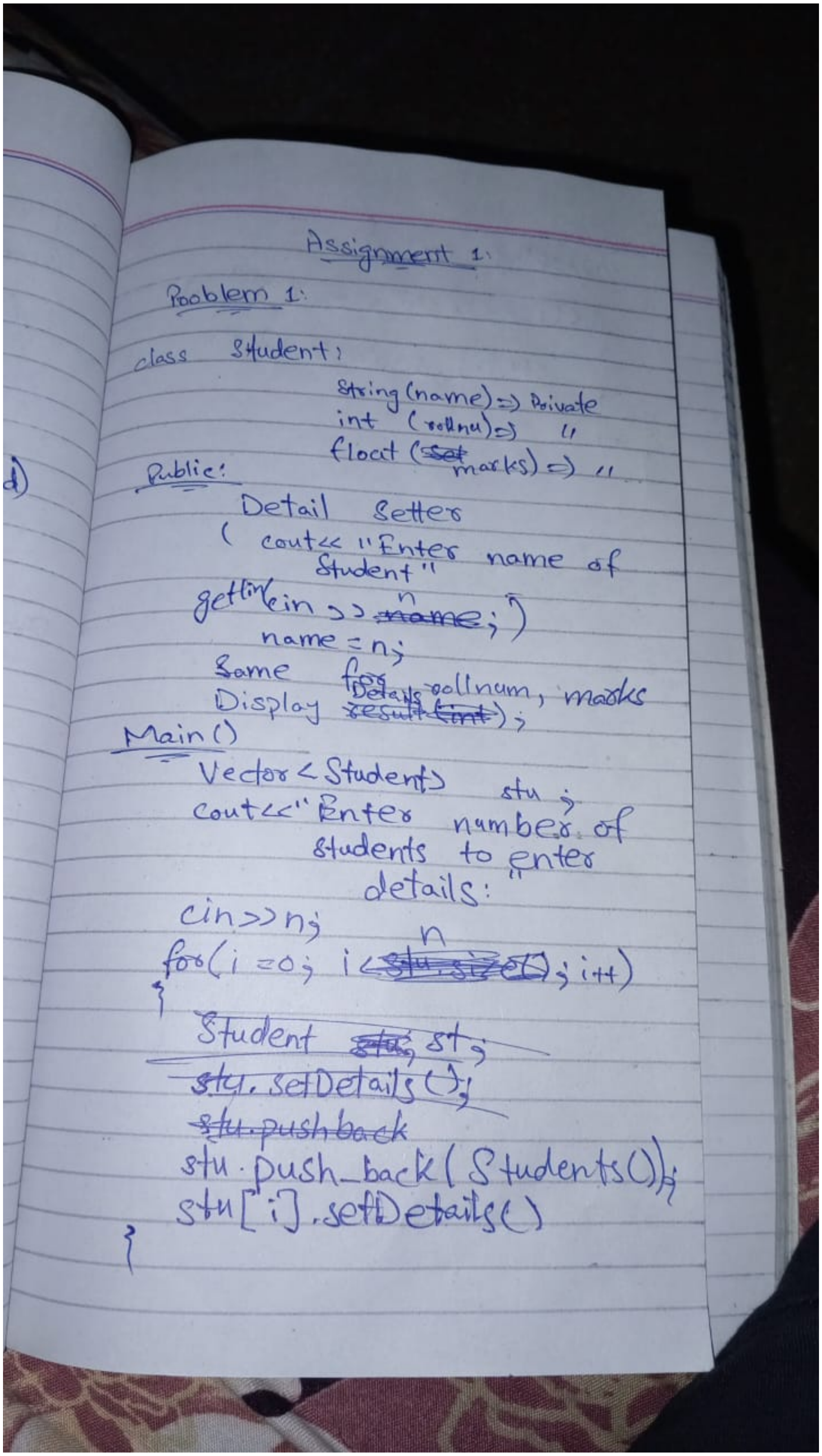
DSA Assignment 1 - (53457) M. Abdullah

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:

Prooblem 1:

class Student:

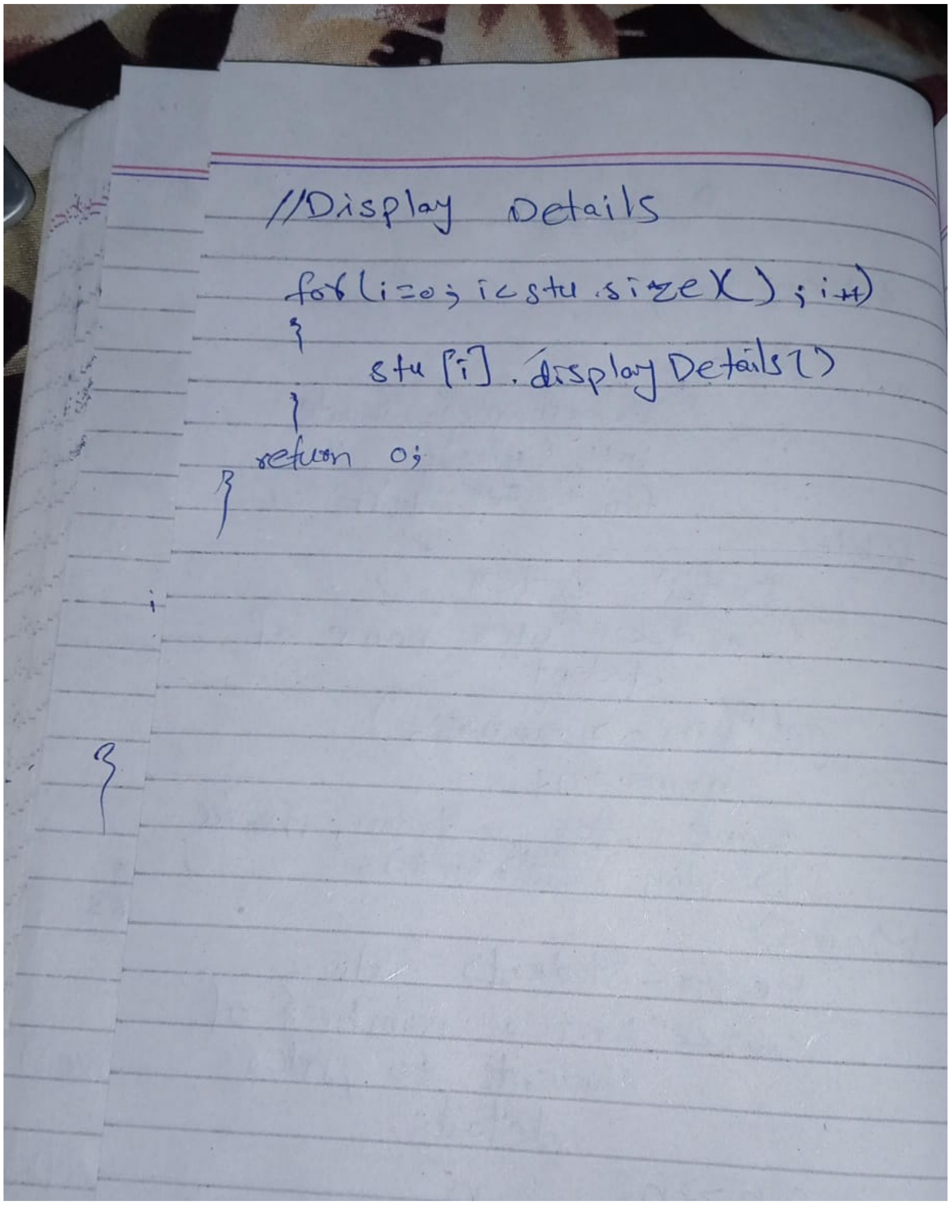
String (name) => Private
int (rollnu) => "
float (~~Set~~ marks) => "

Public:

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

Main()

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

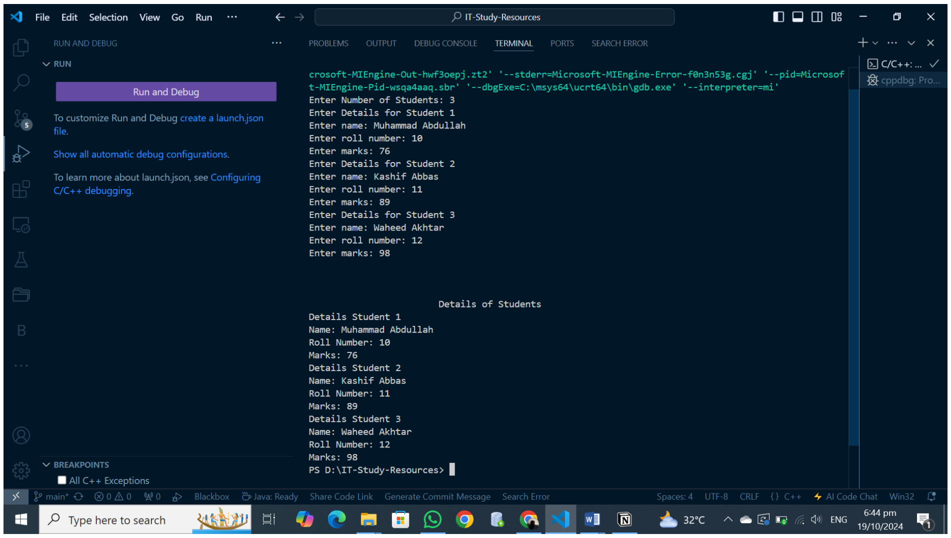


```
//Display Details

for(i=0; i<stu.size(); i++)
{
    stu[i].displayDetails()
}
return 0;
}
```

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



```
crosoft-MIEngine-Out-hwf3oepj.zt2' '--stderr=Microsoft-MIEngine-Error-f0n3n53g.cgj' '--pid=Microsof
t-MIEngine-Pid-wsqa4aaq.sbr' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter Number of Students: 3
Enter Details for Student 1
Enter name: Muhammad Abdullah
Enter roll number: 10
Enter marks: 76
Enter Details for Student 2
Enter name: Kashif Abbas
Enter roll number: 11
Enter marks: 89
Enter Details for Student 3
Enter name: Waheed Akhtar
Enter roll number: 12
Enter marks: 98



                        Details of Students
Details Student 1
Name: Muhammad Abdullah
Roll Number: 10
Marks: 76
Details Student 2
Name: Kashif Abbas
Roll Number: 11
Marks: 89
Details Student 3
Name: Waheed Akhtar
Roll Number: 12
Marks: 98
PS D:\IT-Study-Resources>
```

File Edit Selection View Go Run ···
IT-Study-Resources
RUN AND DEBUG
RUN
Run and Debug
To customize Run and Debug create a launch.json file.
Show all automatic debug configurations.
To learn more about launch.json, see Configuring C/C++ debugging.
BREAKPOINTS
All C++ Exceptions
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR
C/C++: ...
cppdbg: Pro...
main* Blackbox Java: Ready Share Code Link Generate Commit Message Search Error Spaces: 4 UTF-8 CRLF {} C++ AI Code Chat Win32
Type here to search
32°C ENG 6:44 pm 19/10/2024

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**.

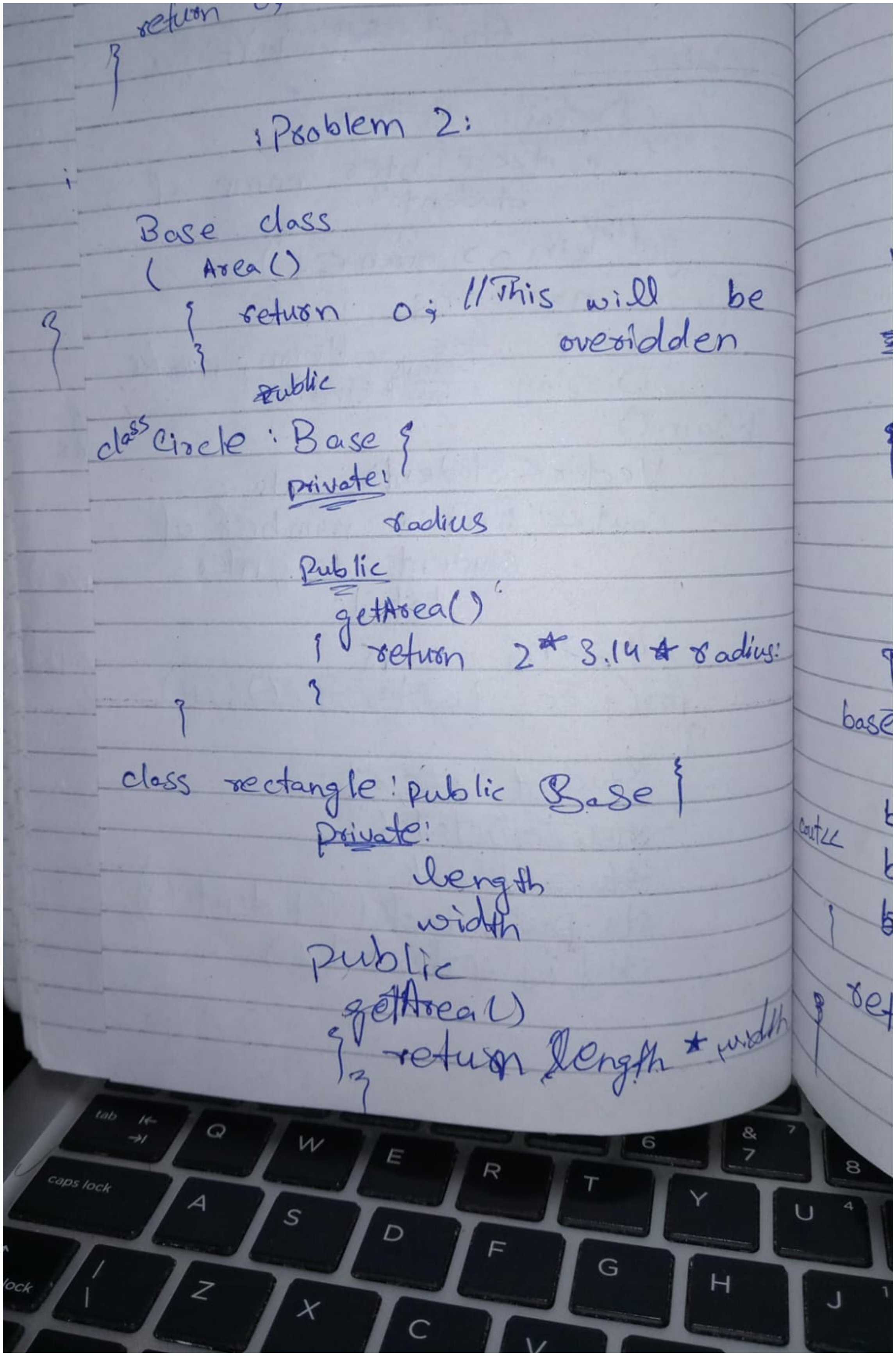
Requirements:

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.

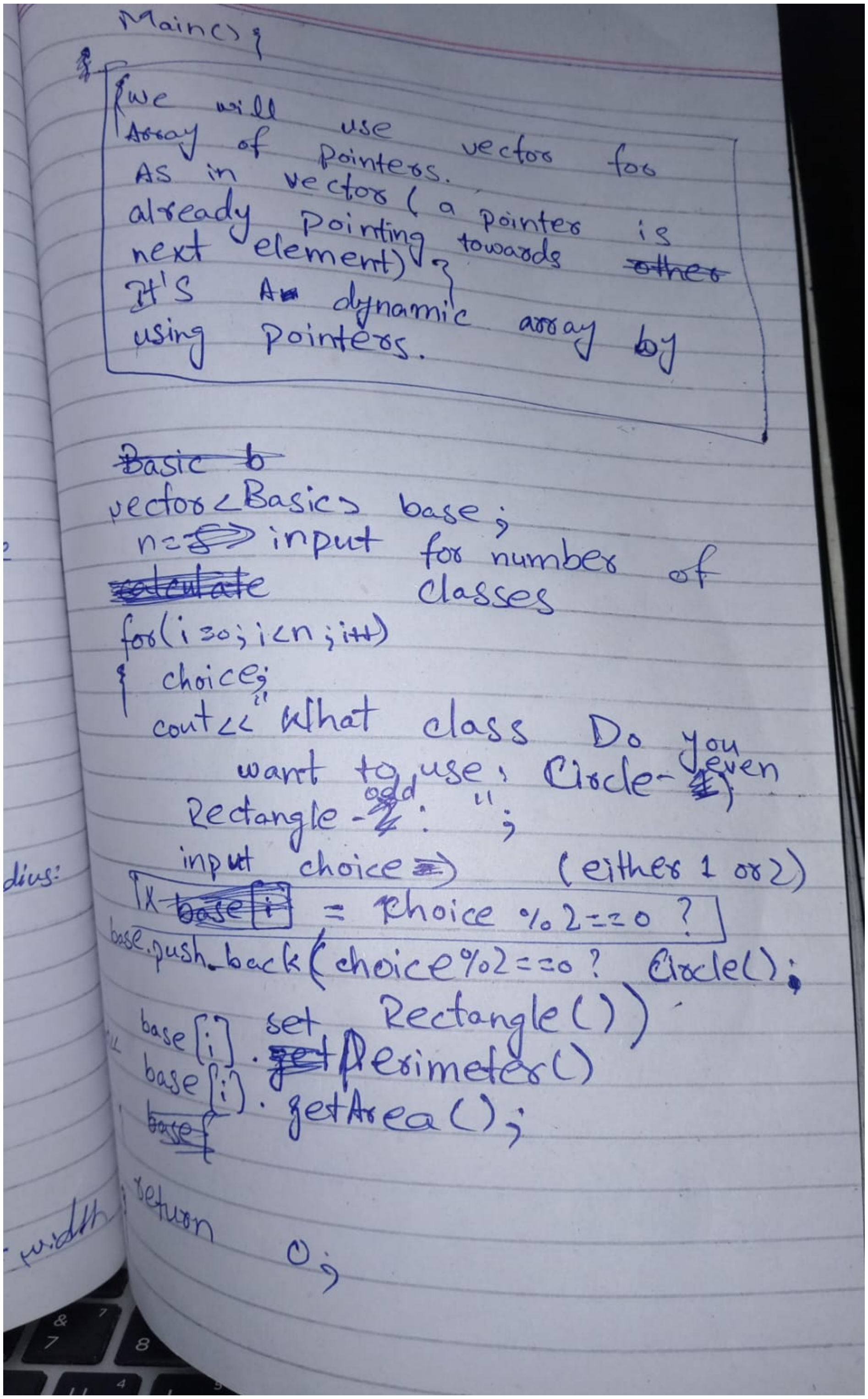
3. **Derived Class Rectangle** (inherits from Shape):
    - Data Members:
        - length (float)
        - width (float)
    - Override getArea() to calculate the area of a rectangle.
4. **Main Task**:
    - Use an array of pointers to the base class to store different shapes.

Calculate and display the area of each shape.

Problem Solving



```
      return 0;
    }
  }

        ;Problem 2:

;

    Base class
    ( Area()
      { return 0; //This will be
      }                  overidden
}
          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
          }
```

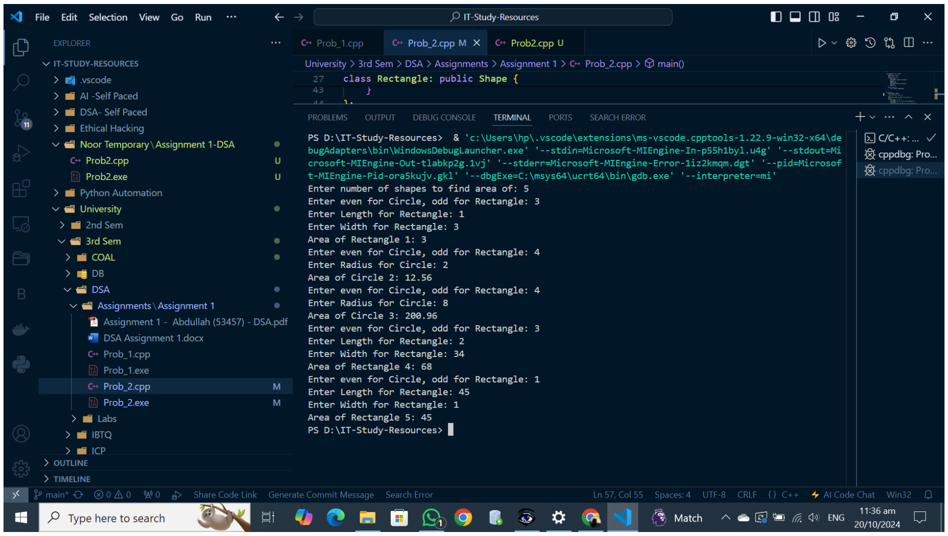


```
Main(){
{
  {We will use vector for
  Array of Pointers.
  As in vector ( a pointer is
  already pointing towards ~~other~~
  next element)}
  It's A dynamic array by
  using pointers.

~~Basic b~~
vector<Basic> base;
  n=⇒ input for number of
~~calculate~~    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)
  [X ~~base[i]~~ = choice %2==0 ?]
base.push_back(choice%2==0 ? Circle():
                Rectangle())
  base[i].set~~get~~Perimeter()
  base[i].getArea();
  ~~base[~~
}
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) {
            shapes.push_back(new Circle());
            ((Circle*)shapes[i])->setPerimeter();
            cout<<"Area of Circle "<<i+1<<": "<<((Circle*)shapes
        }
        else {
            shapes.push_back(new Rectangle());
            ((Rectangle*)shapes[i])->setPerimeter();
            cout<<"Area of Rectangle "<<i+1<<": "<<((Rectangle*)
        }
    }
    /*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\t\n";
    for(int i=0;i<rectangles.size();i++) {
        cout<<"Area of Rectangle "<<i+1<<": "<<rectangles[i].get
    }*/
    return 0;
}
```

Output



```
class Rectangle: public Shape {
    }
```

```
PS D:\IT-Study-Resources>  & 'c:\Users\hp\.vscode\extensions\ms-vscode.cpptools-1.22.9-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-p55h1byl.u4g' '--stdout=Microsoft-MIEngine-Out-tlabkp2g.1vj' '--stderr=Microsoft-MIEngine-Error-1iz2kmqm.dgt' '--pid=Microsoft-MIEngine-Pid-ora5kujv.gkl' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Enter number of shapes to find area of: 5
Enter even for Circle, odd for Rectangle: 3
Enter Length for Rectangle: 1
Enter Width for Rectangle: 3
Area of Rectangle 1: 3
Enter even for Circle, odd for Rectangle: 4
Enter Radius for Circle: 2
Area of Circle 2: 12.56
Enter even for Circle, odd for Rectangle: 4
Enter Radius for Circle: 8
Area of Circle 3: 200.96
Enter even for Circle, odd for Rectangle: 3
Enter Length for Rectangle: 2
Enter Width for Rectangle: 34
Area of Rectangle 4: 68
Enter even for Circle, odd for Rectangle: 1
Enter Length for Rectangle: 45
Enter Width for Rectangle: 1
Area of Rectangle 5: 45
PS D:\IT-Study-Resources>
```