**Worksheet-2**

**Student Name:-** Pushpraj Roy **UID:-** 20BCS9866

**Branch:-** BE- CSE **Section/Group:-** WM\_617 “A”

**Subjetct Code:-** 20CSP-312 **Semester:-** 5th

**Subject Name:-** DAA Lab

1. **Aim/Overview of the practical: -**

The code implements a power function in O(log n) time.

# Task to be done/ Which logistics used :-

To find Power of a number

1. **Algorithm/Flowchart :-**

Step 1- Take x and n input.

Step 2- Calculate pow(x, n) method check base condition if n==0 return 1 check base condition if

n==1 return x recursively call pow(x,n-1) and go to step 2;

Step 3- Print result.

1. **Steps for experiment/practical/Code :-**

#include<iostream>

using namespace std;

int power(int x, int y)

{

if (y == 0)

return 1;

else if (y % 2 == 0)

return power(x, y / 2) \* power(x, y / 2);

else

return x \* power(x, y / 2) \* power(x, y / 2);

}

int main()

{

int x,y;

cout<<"Enter numbers:"<<endl;

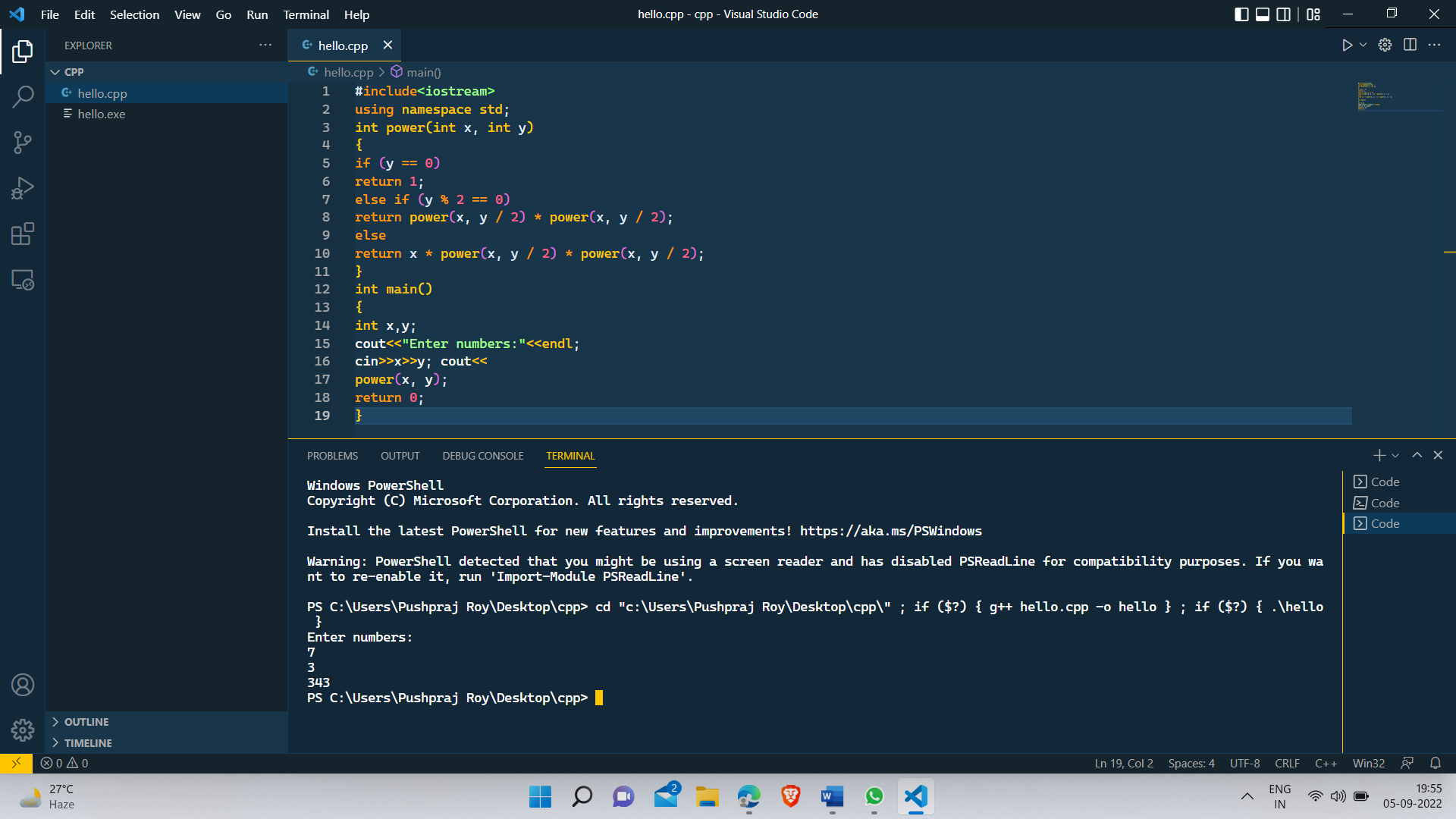
cin>>x>>y; cout<<

power(x, y);

return 0;

}

1. **Result/Output :-**

****

1. **Observations/Discussions/ Complexity Analysis :-**

Time complexity of finding power of a number using recursion is O(log n).

1. **Learning outcomes (What I have learnt) :-**

* To know to calculate power of a function.
* To learn how to use recursion for solving problems.

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |
|  |  |  |  |