

**Experiment: 2**

**Student Name:** Sahil Kaundal **UID:** 21BCS8197

**Branch:** BE CSE (Lateral Entry) **Section/Group:** 616/A

**Semester:** 5th **Date of Performance:** 05/09/2022

**Subject Name:** DAA Lab **Subject Code:** 21-CSP-312

# Aim/Overview of the practical:

# Code implement power function in O(logn) time complexity.

# Task to be done/ Which logistics used:

To find Power of a number.

1. **Algorithm/Flowchart:**

***Step1:*** Take x and n input.

***Step2:*** Calculate pow(x, n) method check base condition if n==0 return 1 check base condition if n==1 return x recursively callpow(x,n-1) and go to step 2;

***Step 3:*** Print result.

# Steps for experiment/practical/Code:

#include<bits/stdc++.h>

using namespace std;

double power(double n,int x)

{

if(x==0) return 1;

double temp=power(n,x/2);

if(x%2==0)

{

return temp\*temp;

}

else if(x>0)

{

return n\*temp\*temp;

}

else

{

return (temp\*temp)/n;

}

}

int main()

{

double n;

int x;

cout<<"Enter First Number: ";

cin>>n;

cout<<"Enter Second Number: ";

cin>>x;

double ans=power(n,x);

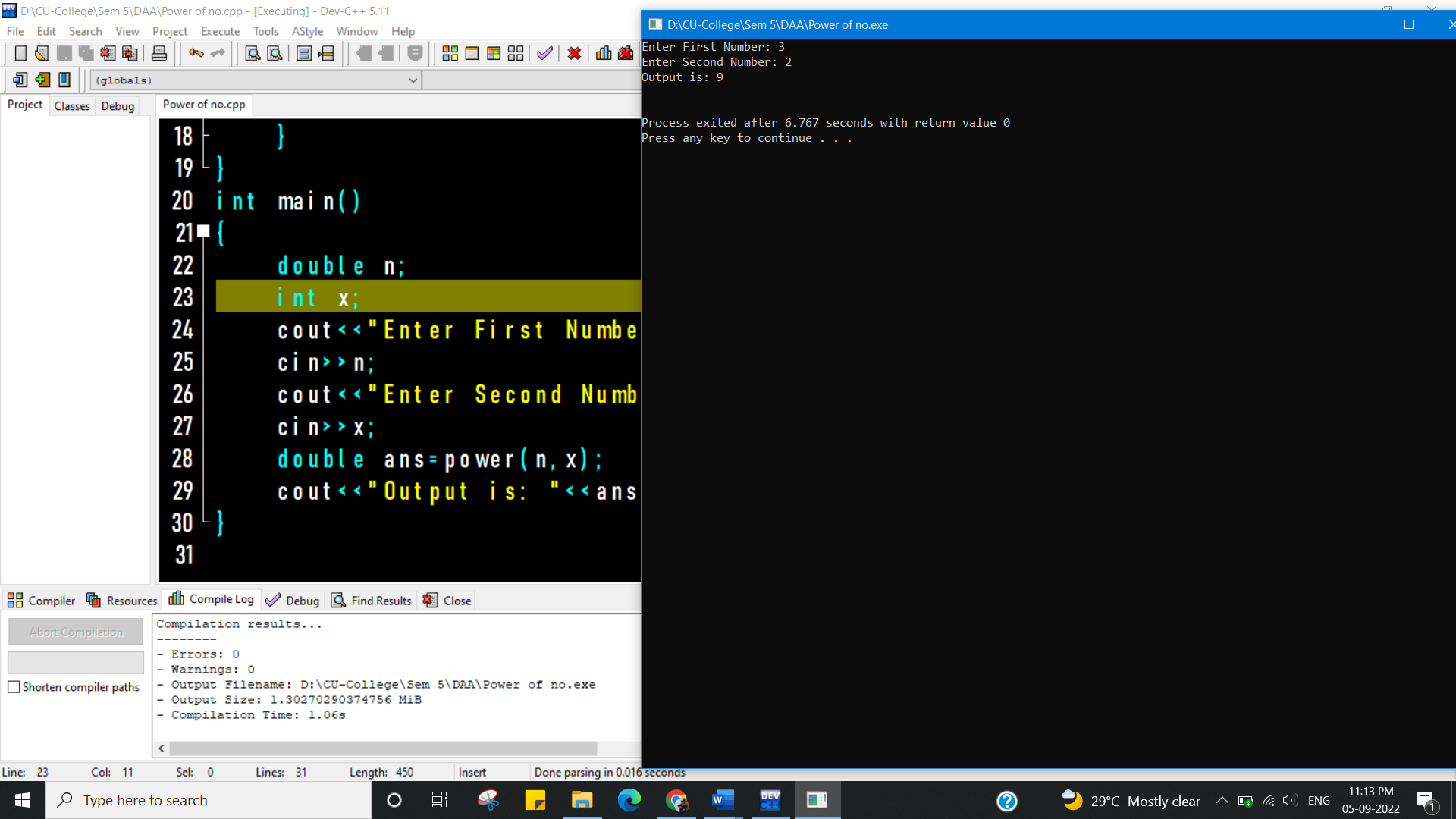
cout<<"Output is: "<<ans<<endl;

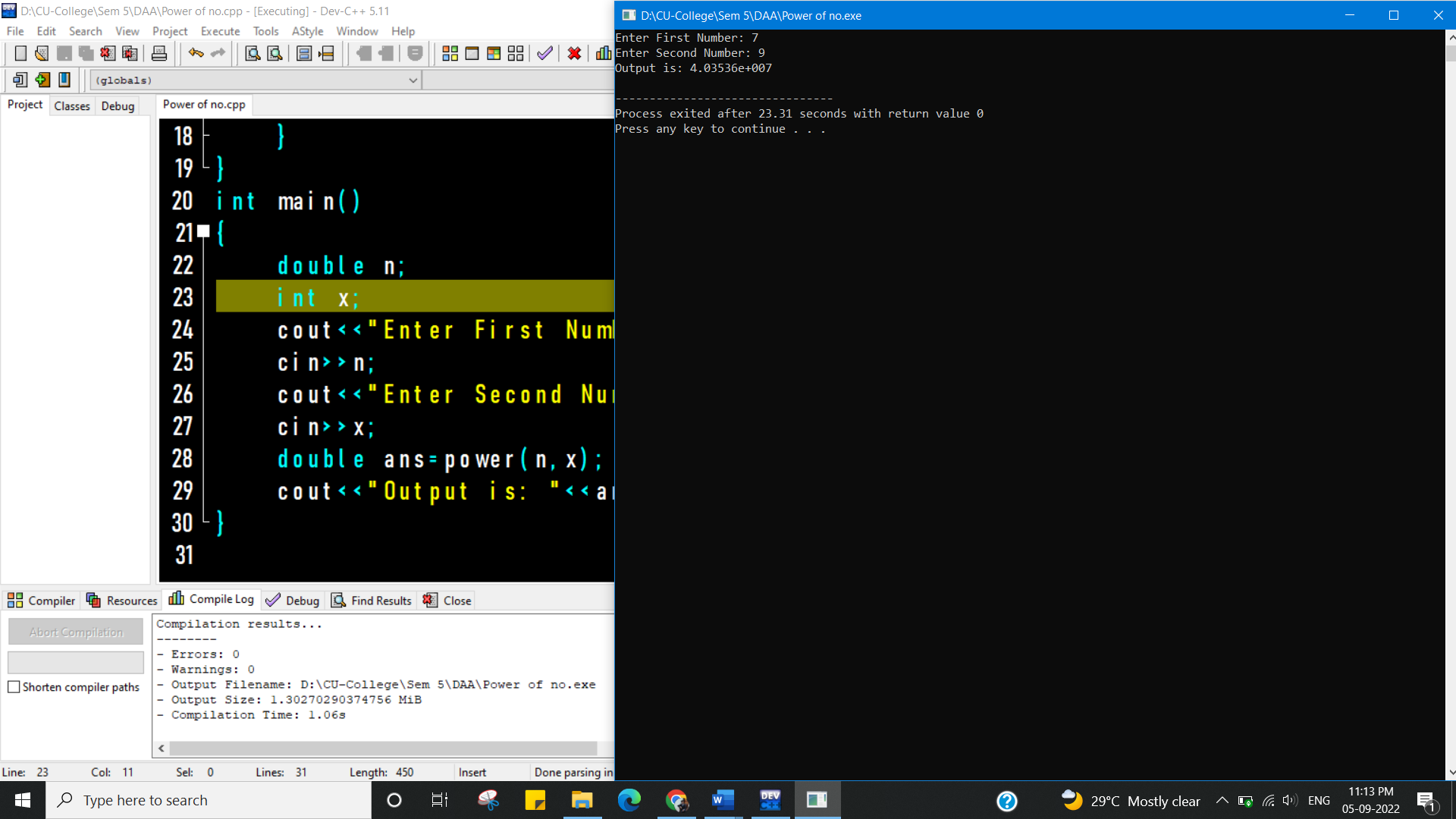
}

# Observations/Discussions/ Complexity Analysis:

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

# Result/Output/Writing Summary:

****

****

**Learning outcomes (What I have learnt):**

1. To know to calculate power of a function.
2. 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. |  |  |  |
|  |  |  |  |