Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Exponential Function with O(N) and O(logN).	
Experiment No: 03	**Date:**	**Enrollment No: 92200133030**

Aim: Exponential Function with O(N) and O(logN)

IDE: Visual Studio Code

I. Exponential Function using Iterative (Naive) Approach :-

Theory: -

This approach calculates the exponential function x^n by multiplying x by itself n times in a loop. Starting from an initial value of 1, each iteration multiplies the result by x, continuing until we've multiplied x n times. This approach is straightforward and easy to implement, but it has a time complexity of O(n) since it requires n multiplications, making it inefficient for large values of n.

Algorithm: -

Programming Language: - C++

Code :-

```
#include <bits/stdc++.h>
using namespace std;

long long Expotential(long base, long power) {
    if (power == 0) {
        return 1;
    }
    if (power == 1) {
        return base;
    }
    return base * Expotential(base, power - 1);
}
```

Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Exponential Function with O(N) and O(logN).	
Experiment No: 03	**Date:**	**Enrollment No: 92200133030**

```
int main() {

    long base;
    long power;
    cout << "Enter the Base of the Ecpotentail :- ";
    cin >> base;
    cout << "Enter the Power of the Expotential :- ";
    cin >> power;

    long long result = Expotential(base, power);

    cout << "The " << base << " raised to " << power << " is " << result << " ." << endl;
    return 0;
}
```

Output :-

```
PS D:\Aryan Data\Usefull Data\Semester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3> cd "d:\Aryan Data\Usefull Data\Sem
ester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3\" ; if ($?) { g++ Expotentail_Using_Naive.cpp -o Expotentail_Using_N
aive } ; if ($?) { .\Expotentail_Using_Naive }
Enter the Base of the Ecpotentail :- 25
Enter the Power of the Expotential :- 5
The 25 raised to 5 is 9765625 .
PS D:\Aryan Data\Usefull Data\Semester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3>
```

Space Complexity:- _____
Justification: -

Time Complexity:

Best Case Time Complexity: _____
Justification: -

Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Exponential Function with O(N) and O(logN).	
Experiment No: 03	**Date:**	**Enrollment No: 92200133030**

Worst Case Time Complexity:- _____
Justification: -

II. Exponential Function with O(N) using Divide and Conquer Approach :-

Theory: -

This recursive method improves upon the naive approach by leveraging the properties of exponentiation through division:

- If n is even, the function computes x^(n/2) and squares the result.
- If n is odd, it computes x^[(n-1)/2], squares it, and then multiplies by x. Using divide and conquer

Algorithm: -

Programming Language: - C++

Code :-

```
#include <bits/stdc++.h>
using namespace std;

long long Expotential(long base, long power) {
    if (power == 0) {
        return 1;
    }
```

Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Exponential Function with O(N) and O(logN).	
Experiment No: 03	**Date:**	**Enrollment No: 92200133030**

```
        else if (power % 2 == 0) {
            return Expotential(base, power / 2) * Expotential(base, power / 2);
        }

        else {
            return base * Expotential(base, power / 2) * Expotential(base, power / 2);
        }
    }

    int main() {

        long base;
        long power;

        cout << "Enter the Base of the Ecpotentail :- ";
        cin >> base;

        cout << "Enter the Power of the Expotential :- ";
        cin >> power;

        long long result = Expotential(base, power);

        cout << "The " << base << " raised to " << power << " is " << result << " ." << endl;
        return 0;
    }
```

Output:-

```
PS D:\Aryan Data\Usefull Data\Semester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3> cd "d:\Aryan Data\Usefull Data\Sem
ester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3\" ; if ($?) { g++ Expotentail_Using_DandC.cpp -o Expotentail_Using_D
andC } ; if ($?) { .\Expotentail_Using_DandC }
Enter the Base of the Ecpotentail :- 25
Enter the Power of the Expotential :- 5
The 25 raised to 5 is 9765625 .
PS D:\Aryan Data\Usefull Data\Semester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3>
```

Space Complexity:- _____

Justification: -

_____

_____

_____

_____

_____

Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Exponential Function with O(N) and O(logN).	
Experiment No: 03	**Date:**	**Enrollment No: 92200133030**

Time Complexity:

Best Case Time Complexity: _____
Justification: -

Worst Case Time Complexity:- _____
Justification: -

III. Exponential Function with O(logN) using Divide and Conquer Approach :-

Theory: -

This optimized divide-and-conquer method also leverages the properties of exponentiation::

- If n is even, the function computes x^(n/2) and squares the result.
- If n is odd, it computes x^[(n-1)/2], squares it, and then multiplies by x. By dividing n by 2 on O(logN). This approach is highly efficient and ideal for large values of n as it minimizes the number of multiplications needed to compute the result.

Algorithm: -

Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Exponential Function with O(N) and O(logN).	
Experiment No: 03	**Date:**	**Enrollment No: 92200133030**

Programming Language: - C++

Code :-

```
#include <bits/stdc++.h>
using namespace std;

long long Expotential(long base, long power) {
    if (power == 0) {
        return 1;
    }
    if (base == 0) {
        return 0;
    }
    if (power < 0) {
        return  1 / Expotential(base, power * -1);
    }
    if (power == 1) {
        return base;
    }
    long long Half_Power = Expotential(base, power / 2);
    if (power % 2 == 0) {
        return Half_Power * Half_Power ;
    }
    else {
        return base * Half_Power * Half_Power ;
    }
}

int main() {

    long base;
    long power;
    cout << "Enter the Base of the Expotentail :- ";
    cin >> base;
    cout << "Enter the Power of the Expotential :- ";
    cin >> power;
    long long result = Expotential(base, power);
    cout << "The " << base << " raised to " << power << " is " << result << " ." << endl;
    return 0;
}
```

Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Exponential Function with O(N) and O(logN).	
Experiment No: 03	**Date:**	**Enrollment No: 92200133030**

Output:-

```
PS D:\Aryan Data\Usefull Data\Semester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3> cd "d:\Aryan Data\Usefull Data\Sem
ester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3\" ; if ($?) { g++ Expotential_Using_DandC_Optimized.cpp -o Expotenti
al_Using_DandC_Optimized } ; if ($?) { .\Expotential_Using_DandC_Optimized }
Enter the Base of the Expotentail :- 25
Enter the Power of the Expotential :- 5
The 25 raised to 5 is 9765625 .
PS D:\Aryan Data\Usefull Data\Semester - 5\Design-and-Analysis-of-Algorithms\Lab - Manual\Experiment - 3>
```

Space Complexity:- _____
Justification: -

Time Complexity:

Best Case Time Complexity: _____
Justification: -

Worst Case Time Complexity:- _____
Justification: -

Conclusion:-