 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: DAA (01CT0512)	AIM: Exponential / Power function of order O(N)	
Experiment No: 9	Date: 29/8/2023	Enrolment No: 92100133020

Exponential / Power function of order O(N):


Exponential/Power function algorithms are characterized by their time complexity growing exponentially with the size of the input (N). These algorithms often involve repeated multiplication or exponentiation.

Algorithm:

The algorithm follows a straightforward approach where each step involves an exponential or power operation on the input.

Code:

```
#include <iostream>
using namespace std;
int power(int x,int n){
    if(n==0){
        return 1;
    }
    int temp= power(x,(n/2));
    if(n%2==0){
        return temp*temp;
    }
    else{
        return x*temp*temp;
    }
    return 0;
}
int power_2(int x,int n){
    if(n==0){
        return 1;
    }
    int temp=1;
    for (int i = 0; i < n; i++){
        temp *= x;
    }
    return temp;
}
int main(){
    int number,power_1;
```

 Marwadi University	Marwadi University Faculty of Technology Department of Information and Communication Technology	
Subject: DAA (01CT0512)	AIM: Exponential / Power function of order O(N)	
Experiment No: 9	Date: 29/8/2023	Enrolment No: 92100133020

```

cout<<"Enter Number :- ";
cin >> number;
cout<<"Enter Power :- ";
cin >> power_1;
cout<<"From first method"<<endl;
cout<< power(number,power_1)<<endl;
cout<<"From second method"<<endl;
cout<< power_2(number,power_1)<<endl;
return 0;}

```

Output:

```

Enter Number :- 5
Enter Power :- 3
From first method
125
From second method
125

```

Space complexity: _____

Justification: _____

Time complexity:

Best case time complexity: _____

Justification: _____

Worst case time complexity: _____

Justification: _____