Algorithms Analysis and Design

Sama Haitham Sammar 202110795

=>Assignment 1

Factorial Calculation in two methods and compare execution time.

Here record of execution time for each test case:-

n=5	Iterative function	Recursive function
Execution time	2.147 microseconds	0.516 microseconds

n=7	Iterative function	Recursive function
Execution time	1.855 microseconds	1.208 microseconds

n=10	Iterative function	Recursive function
Execution time	2.076 microseconds	0.799 microseconds

n=16	Iterative function	Recursive function
Execution time	1.99 microseconds	1.64 microseconds

n=20	Iterative function	Recursive function
Execution time	1.852 microseconds	0.708 microseconds

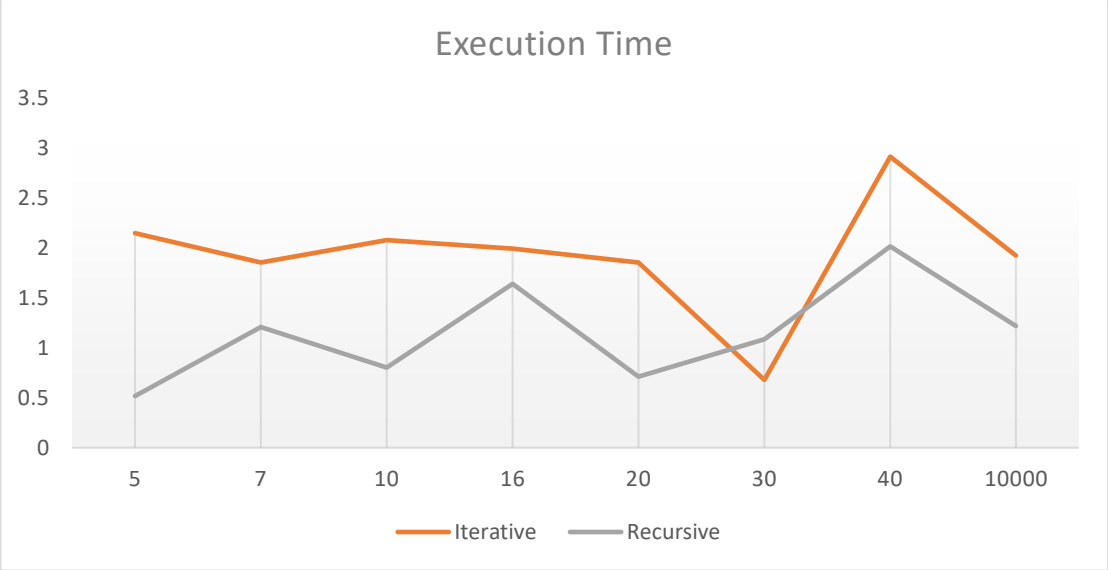
=> In Iterative :The results when the value of n is more than 20 are NEGATIVE and if the number large more the result ZERO, because there is a maximum size can hold of variable type in c++ ,also when the result is large numbers, this type(long long) cannot carry it.

n=30	Iterative function	Recursive function
Execution time	0.678 microseconds	1.085 microseconds

n=40	Iterative function	Recursive function
Execution time	2.912 microseconds	2.014 microseconds

n=10000	Iterative function	Recursive function
Execution time	1.921 microseconds	1.22 microseconds

=> In Recursive :There is no result of 10000! (large numbers) because of STACK OVERFLOW this happens when program try to use more memory in the stack than has been allocated , so this makes an error .



Here is a chart showing the time for each method.

Conclusions :-

1- At first, I noticed that the Recursive time was increasing, and this is clear in the chart.

2- Also when stack overflow occurs and its cause was previously explained.

3- Recursive solution requires extra space for the call stack, so the Iterative in this problem the more efficient choice.

4- The iteration function runs much faster and more space efficient than the recursive, the Iterative takes almost constant time.

```cpp
#include <iostream>
#include <chrono>
using namespace std;
using namespace chrono;
//Iterative function to find factorial of a number.
long long iterativeFactorial( long long n)
{
    long long fact =1;
    for( int i=1; i<=n; i++)
    {
        fact=fact*i;
    }
    return fact;
}
//Recursive function to find factorial of a number.
long long recursiveFactorial (long long n)
{
    if (n==0)
    {
        return 1;
    }
    return n*recursiveFactorial(n-1);
}
int main()
{
    long long n;
    cout<<"Inter a number n to find n!"<<endl;
    cin>>n;

    auto start = high_resolution_clock::now();

    cout<<"The Factorial of n by Iterative Method is "
<<iterativeFactorial(n)<<endl;

  //cout<<"The Factorial of n by Recursive Method is "
<<recursiveFactorial(n)<<endl;

    auto finish = high_resolution_clock::now();

    // Calculating execution time taken by the program.
    auto duration= duration_cast<microseconds>(finish - start);
    cout << "Time taken in microseconds : "<<
(double)(duration.count() / 1000.0) << endl;
    return 0;
}
```

Here is the code in c++.