Thursday, November 9, 2023

Report :

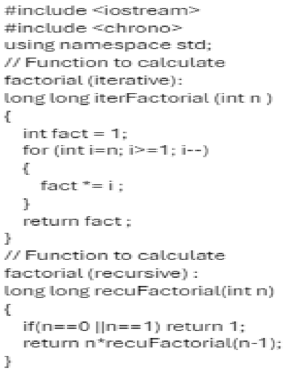
1- A table to show the relationship between execution time results and different functions:

	Iterative	**Recursive**
Values	3	3
execution time	2.29e-07 sec	2.81e-07 sec
Values	5	5
execution time	4.28e-07 sec	3.03e-07 sec
Values	11	11
execution time	1.67e-07 sec	2.18e-07 sec
Values	20	20
execution time	2.04e-07 s * no correct answer	3.52e-07 sec
Values	30	30
execution time	1.85e-07 sec	4.51e-07 sec
Values	50	50
execution time	2.87e-07 sec *no answer	3.52e-07 sec * no correct answer
Values	100	100
execution time	6.83e-07 sec *no answer	1.214e-06 sec *no answer

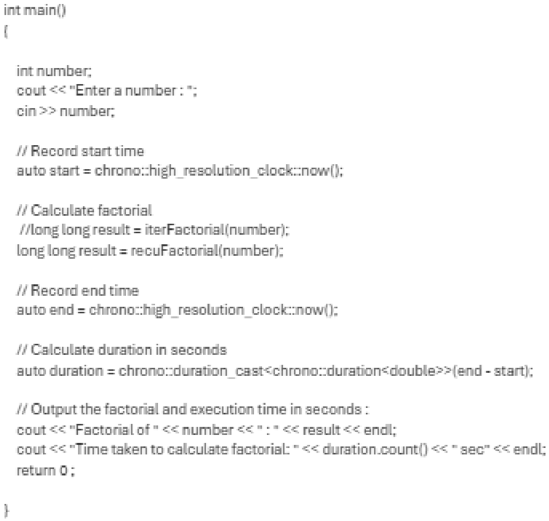
2 - Stack Overflow Proplem And some Note :

"When entering multiple values, I noticed that we did not get the correct answer for large numbers in both cases because the factorial value of the number exceeds the maximum value that can be stored in the variable, which leads to a loss of accuracy and obtaining incorrect values. This problem can be dealt with using advanced methods, such as using libraries that work with large numbers with high precision, or using mathematical approximations to reduce the size of the calculations. In some cases where no result is obtained at all because a memory overflow problem occurs, it can be solved in several ways, including using a loop or changing and optimizing the memory. However, it is not easy, and the tail recursion method can be used, which is effective in some cases. The time was also variable and imprecise as it depended on the device type and input speed."

3 - Code :



```
#include <iostream>
#include <chrono>
using namespace std;
// Function to calculate
factorial (iterative):
long long iterFactorial (int n )
{
   int fact = 1;
   for (int i=n; i>=1; i--)
   {
      fact *= i ;
   }
   return fact ;
}
// Function to calculate
factorial (recursive) :
long long recuFactorial(int n)
{
   if(n==0 ||n==1) return 1;
   return n*recuFactorial(n-1);
}
```



```
int main()
{

  int number;
  cout << "Enter a number : ";
  cin >> number;

  // Record start time
  auto start = chrono::high_resolution_clock::now();

  // Calculate factorial
   //long long result = iterFactorial(number);
  long long result = recuFactorial(number);

  // Record end time
  auto end = chrono::high_resolution_clock::now();

  // Calculate duration in seconds
  auto duration = chrono::duration_cast<chrono::duration<double>>(end - start);

  // Output the factorial and execution time in seconds :
  cout << "Factorial of " << number << " : " << result << endl;
  cout << "Time taken to calculate factorial: " << duration.count() << " sec" << endl;
  return 0 ;

}
```

Student ID :202110963 ;