

main.cpp

```
1 /*****
2 * AUTHOR      :Faris Hijazi
3 * STUDENT ID  :1039438
4 * LAB #12     :Recursion Performance
5 * CLASS      :CS1B
6 * SECTION    :MW: 7:30pm
7 * DUE DATE   :4/30/19
8 *****/
9
10 #include "header.h"
11
12 int main()
13 {
14     int n;                //IN - number to calculate fib of factorial
15     int i;                //CALC - LCV in for loop
16     int menuOpt;          //IN&CALC - menu option user chooses
17     int numEx;            //CALC - LCV in for loop, num of executions
18                           //      - when calculation ex time
19     high_resolution_clock::time_point t1; //CALC - time before execution
20     high_resolution_clock::time_point t2; //CALC - time after execution
21     long long duration1;    //CALC&OUT - difference between t1 and t2 in
22                           //      - microseconds
23
24     PrintHeader(cout, 'A', "Recursion Performance", 4, "Faris Hijazi");
25
26     menuOpt = menuInput();
27
28     while(menuOpt != 0)
29     {
30         switch (menuOpt)
31         {
32             case EXIT:
33                 break;
34
35             case FAC:
36                 cout << endl << "Enter a number n: ";
37                 cin >> n;
38                 cout << "calculating...\n";
39                 cout << "Factorial of " << n << " is: " << factorialR(n) << endl;
40                 break;
41
42             case FIB:
43                 cout << endl << "Enter a number n: ";
44                 cin >> n;
45                 cout << "Fibonacci series: ";
46                 for(i=0; i < n; i++)
47                 {
48                     cout << endl << fibR(i);
49                     if(i < n-1)
50                     {
51                         cout << ',';
```

```

52         }
53     }
54     cout << endl;
55     break;
56 case FACP:
57     cout << endl << "Enter a number n: ";
58     cin >> n;
59     cout << endl;
60
61     cout << "Measuring execution time for recursive...\n";
62
63     t1 = high_resolution_clock::now();
64     for(numEx=0; numEx<=100; numEx++)
65     {
66         factorialR(n);
67     }
68     t2 = high_resolution_clock::now();
69     duration1 = duration_cast<microseconds>( t2 -t1 ).count();
70
71     cout << "It took the program "<< duration1 << " microseconds to execute.
72     \n\n";
73
74     cout << "Measuring execution time for non recursive...\n";
75     t1 = high_resolution_clock::now();
76     for(numEx=0; numEx<=100; numEx++)
77     {
78         factorial(n);
79     }
80     t2 = high_resolution_clock::now();
81     duration1 = duration_cast<microseconds>( t2 -t1 ).count();
82
83     cout << "It took the program "<< duration1 << " microseconds to execute.
84     \n";
85
86     break;
87 case FIBP:
88     cout << endl << "Enter a number n: ";
89     cin >> n;
90     cout << endl;
91
92     cout << "Measuring execution time for recursive...\n";
93
94     t1 = high_resolution_clock::now();
95     for(numEx=0; numEx<=100; numEx++)
96     {
97         for(i=0; i < n; i++)
98         {
99             fibR(i);
100         }

```

main.cpp

```
101         t2 = high_resolution_clock::now();
102         duration1 = duration_cast<microseconds>( t2 - t1 ).count();
103
104         cout << "It took the program "<< duration1 << " microseconds to execute.
        \n\n";
105
106         cout << "Measuring execution time for non recursive...\n";
107         t1 = high_resolution_clock::now();
108         for(numEx=0;numEx<=100;numEx++)
109         {
110             fib(n);
111         }
112         t2 = high_resolution_clock::now();
113         duration1 = duration_cast<microseconds>( t2 -t1 ).count();
114
115         cout << "It took the program "<< duration1 << " microseconds to execute.
        \n";
116         break;
117
118     }
119     menuOpt = menuInput();
120 }
121
122 return 0;
123 }
```