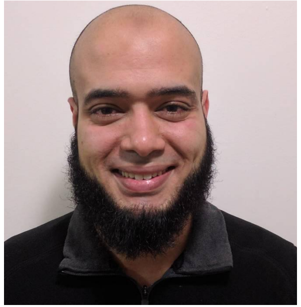
C++ Programming

Recursive Functions 1

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)



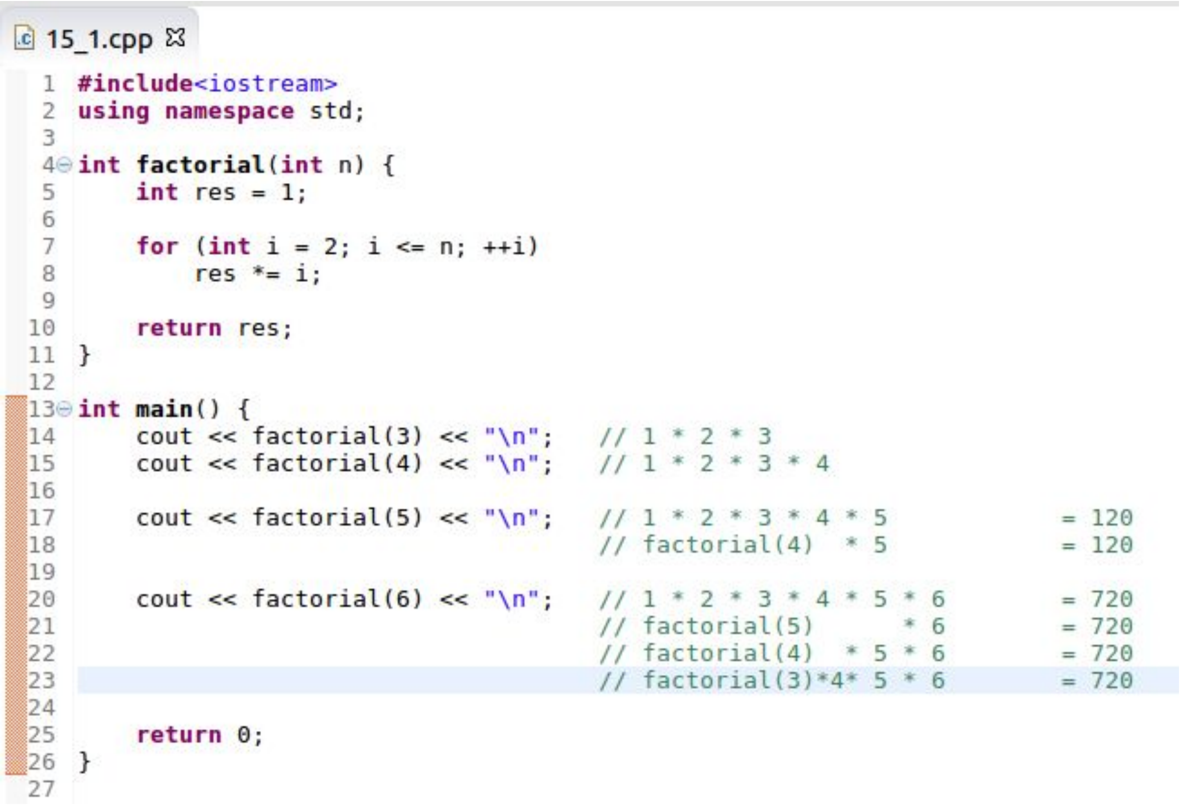
Problem and subproblems

- Sometimes we can decompose a problem to set of sub-problems
- E.g. Print all prime numbers that are palindrome and < 1000000
- We have 2 sub-problems
  - bool is_prime(int n)
  - bool is_palindrome(int n)
- Now we iterate from 1 to 1000000
  - If number satisfy the 2 conditions: count it
- What if the sub-problem is same type as the problem? Recursion!

Recall the factorial

- factorial(6) = **1 * 2 * 3 * 4 * 5** * 6
- factorial(5) = **1 * 2 * 3 * 4 * 5**
- factorial(4) = 1 * 2 * 3 * 4
- factorial(3) = 1 * 2 * 3
- factorial(2) = 1 * 2
- factorial(1) = 1
- Think for a few minutes:
  - What is relation between factorial(6) and factorial(5)?
  - Can you know factorial(6) if you know factorial(5)?

Factorial

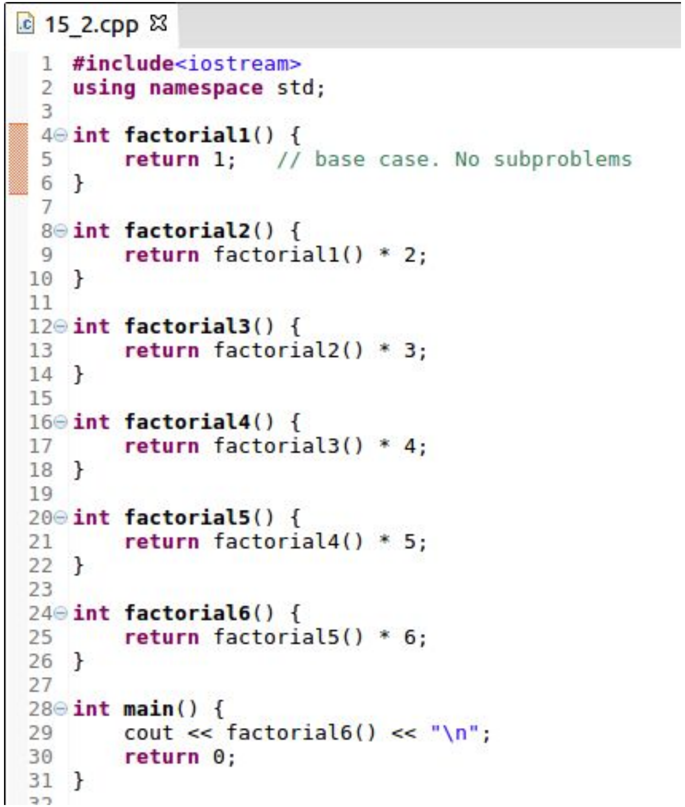


```cpp
#include<iostream>
using namespace std;

int factorial(int n) {
    int res = 1;

    for (int i = 2; i <= n; ++i)
        res *= i;

    return res;
}

int main() {
    cout << factorial(3) << "\n";   // 1 * 2 * 3
    cout << factorial(4) << "\n";   // 1 * 2 * 3 * 4

    cout << factorial(5) << "\n";   // 1 * 2 * 3 * 4 * 5            = 120
                                    // factorial(4)  * 5            = 120

    cout << factorial(6) << "\n";   // 1 * 2 * 3 * 4 * 5 * 6        = 720
                                    // factorial(5)      * 6        = 720
                                    // factorial(4)  * 5 * 6        = 720
                                    // factorial(3)*4* 5 * 6        = 720

    return 0;
}

```

Factorial: Problem and subproblem

- Let say we want to solve factorial(6)
  - This is our problem
  - We can solve it directly with 1*2*3*4*5*6
- Another thinking is: can we think of it is
  - What is factorial(5)? A simpler subproblem
  - Would it help if u know its answer? Yes: 6 * factorial(5) = factorial 6
  - Same logic for factorial(5). It is 5 * factorial(4).
- Going for ever in smaller sub-problems? No
  - There must be a case where no more subproblems. We call it basecase
  - Factorial 1 = 1

Factorial: Problem and subproblem



15_2.cpp

```cpp
#include<iostream>
using namespace std;

int factorial1() {
	return 1;   // base case. No subproblems
}

int factorial2() {
	return factorial1() * 2;
}

int factorial3() {
	return factorial2() * 3;
}

int factorial4() {
	return factorial3() * 4;
}

int factorial5() {
	return factorial4() * 5;
}

int factorial6() {
	return factorial5() * 6;
}

int main() {
	cout << factorial6() << "\n";
	return 0;
}
```

“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”