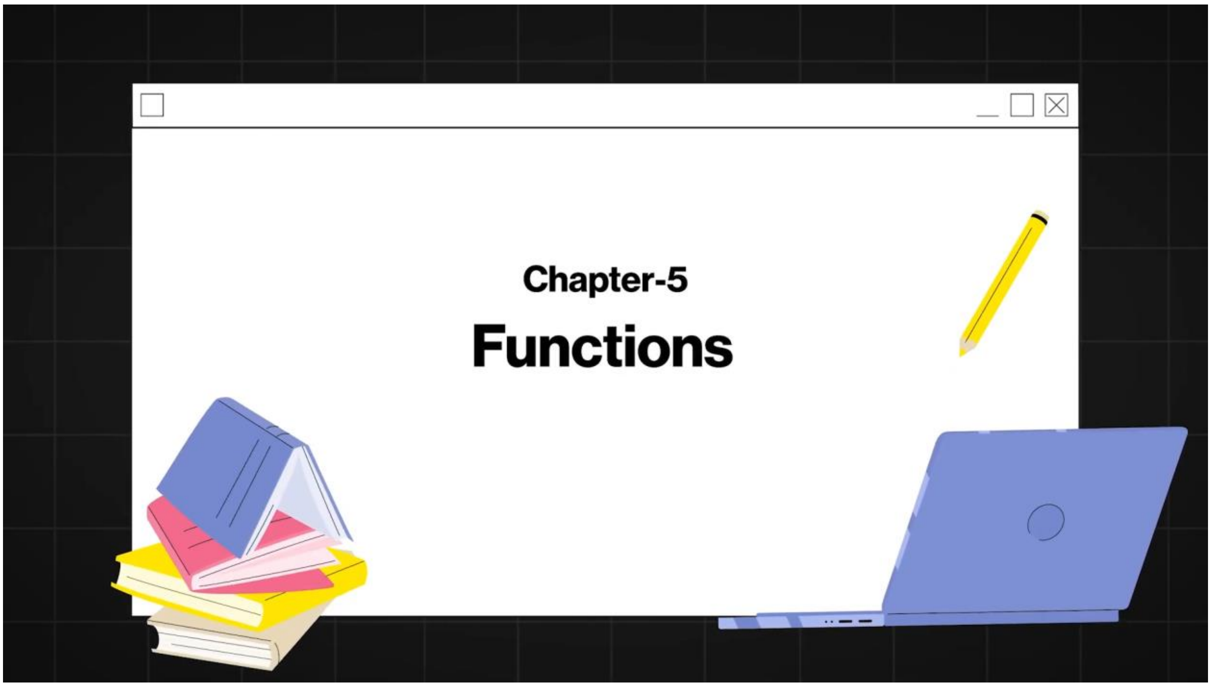
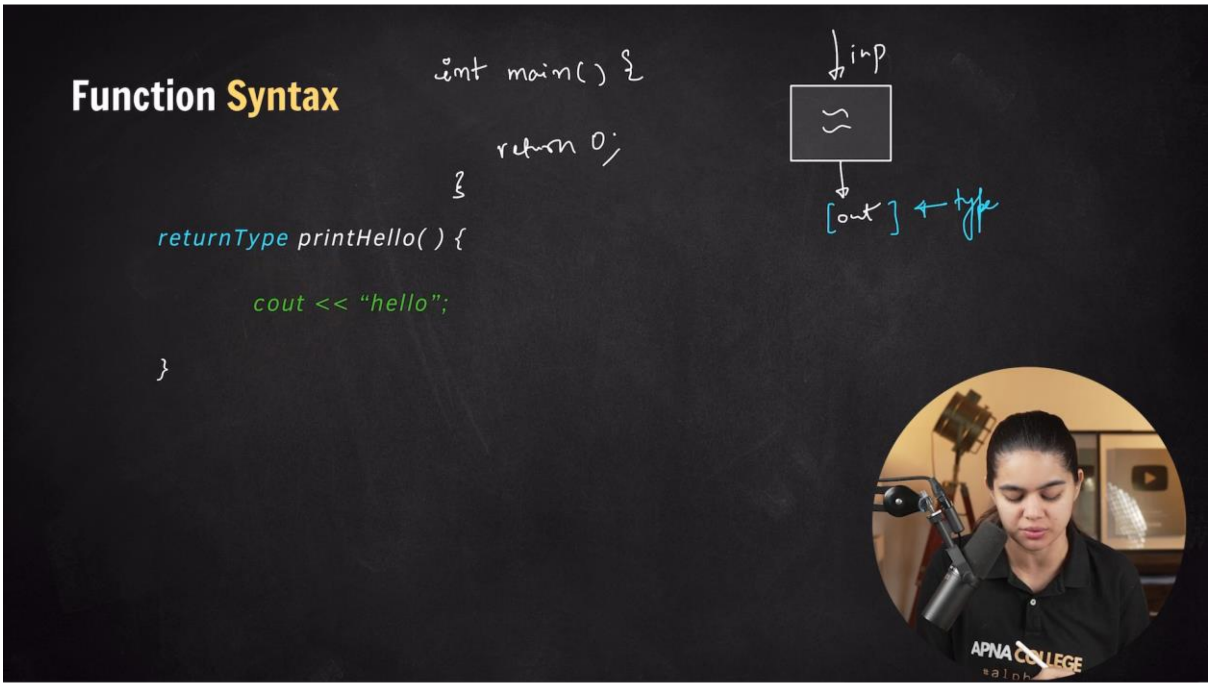


Chapter-5

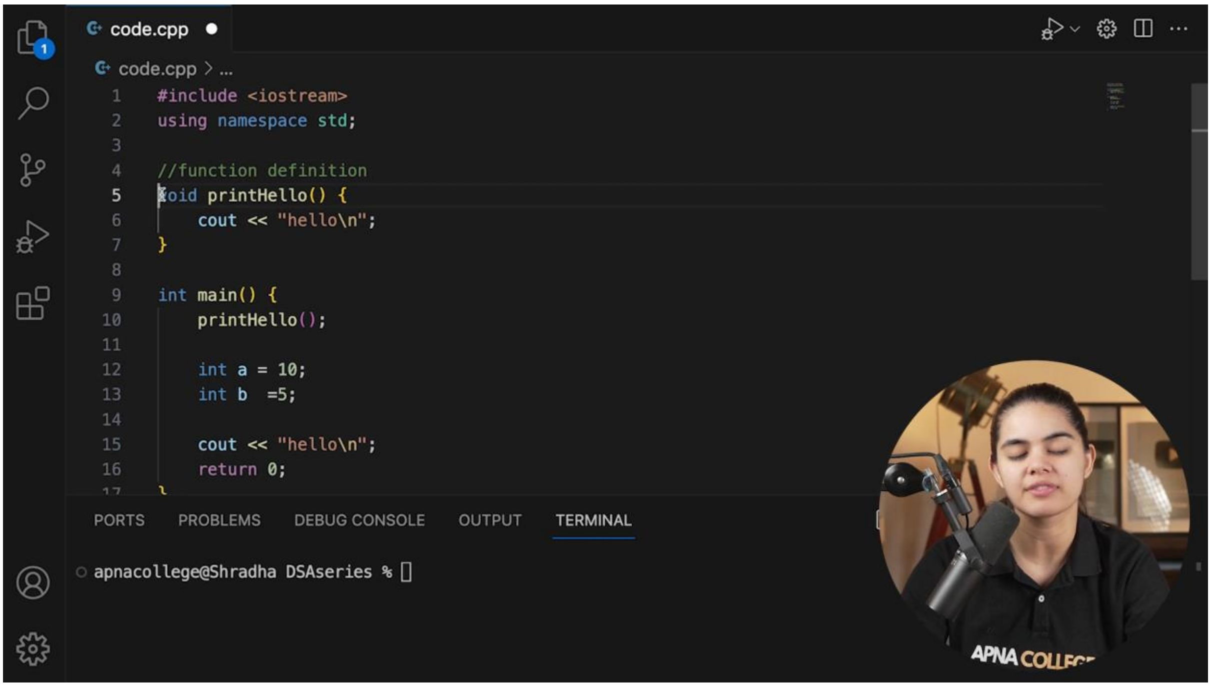
Functions



Function Syntax

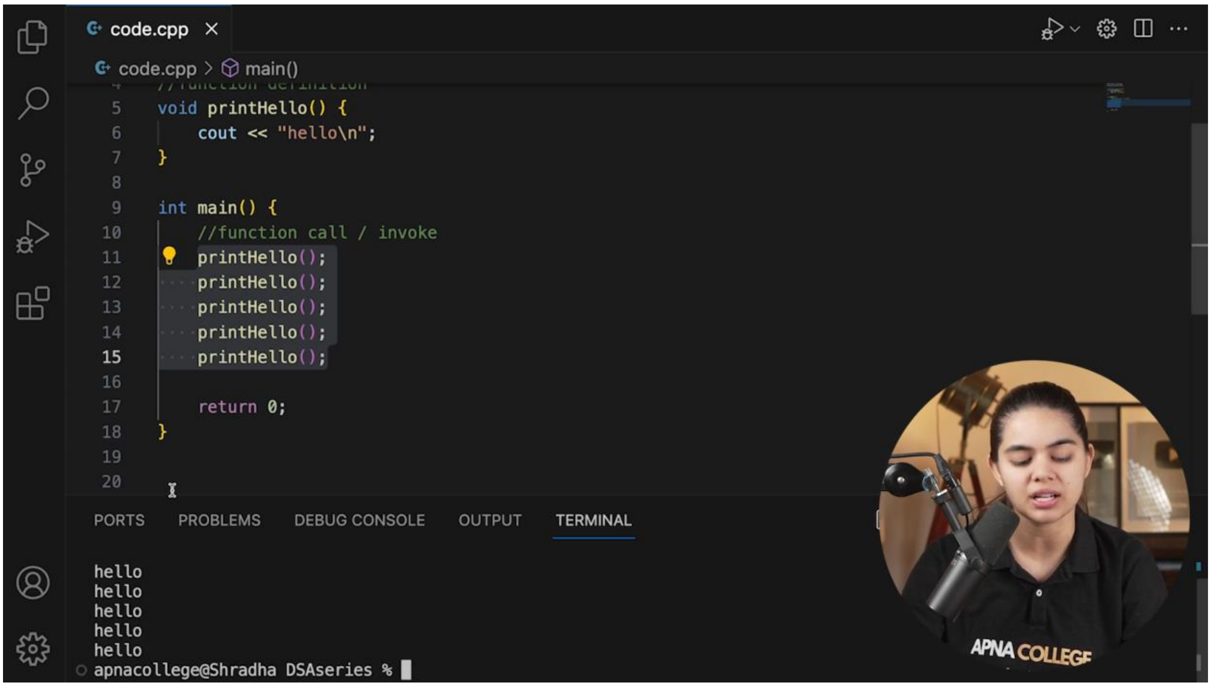
```
int main() {

    return 0;
}
```

```
returnType printHello( ) {

        cout << "hello";

}
```



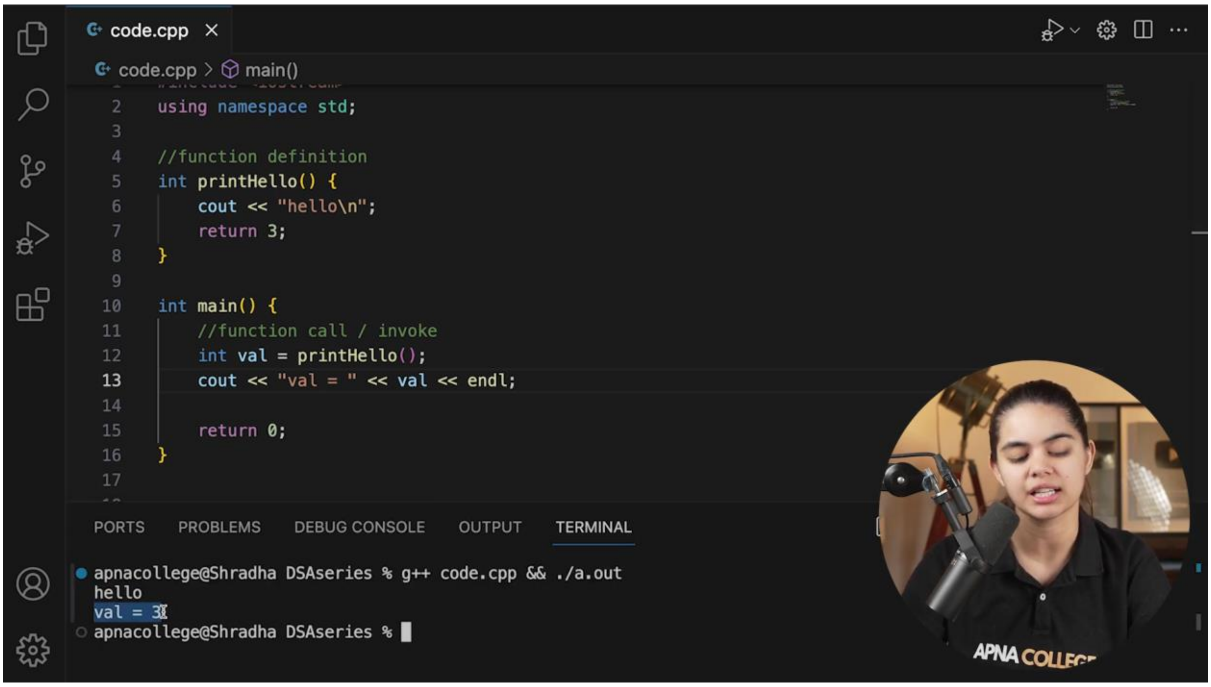
```
#include <iostream>
using namespace std;

//function definition
void printHello() {
    cout << "hello\n";
}

int main() {
    printHello();

    int a = 10;
    int b  =5;

    cout << "hello\n";
    return 0;
}
```

```
apnacollege@Shradha DSAseries % 
```



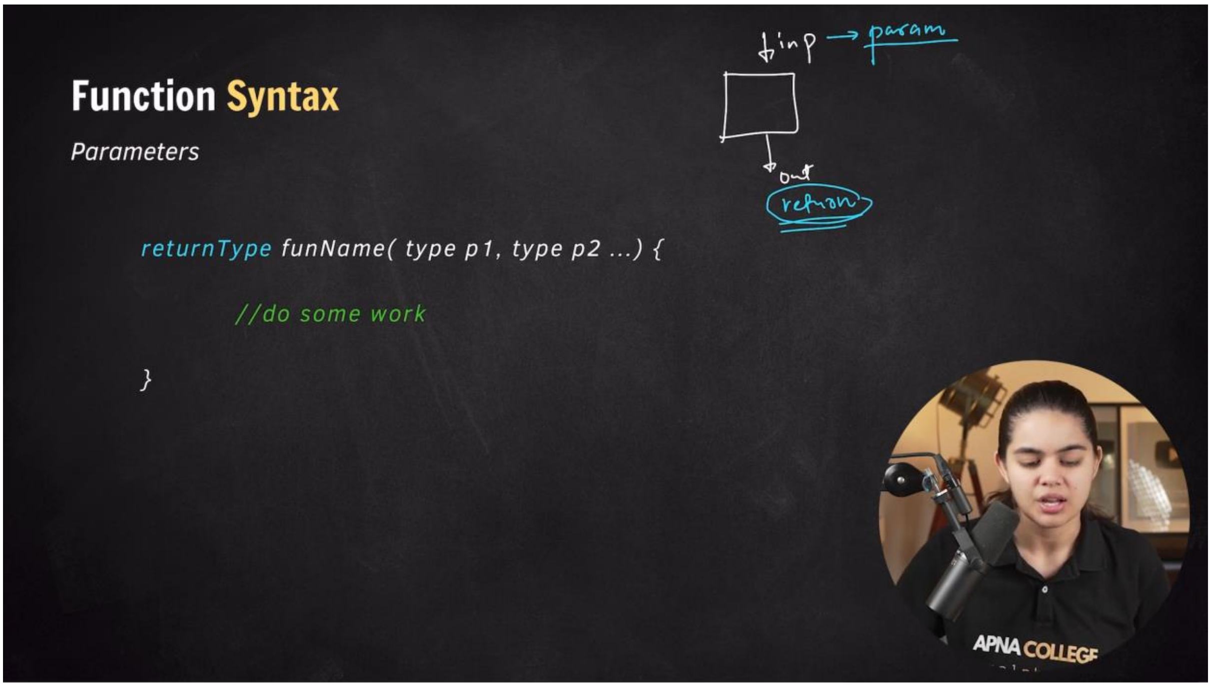
```cpp
//function definition
void printHello() {
    cout << "hello\n";
}

int main() {
    //function call / invoke
    printHello();
    printHello();
    printHello();
    printHello();
    printHello();

    return 0;
}
```

```
hello
hello
hello
hello
hello
apnacollege@Shradha DSAseries % 
```



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

//function definition
int printHello() {
    cout << "hello\n";
    return 3;
}

int main() {
    //function call / invoke
    int val = printHello();
    cout << "val = " << val << endl;

    return 0;
}
```

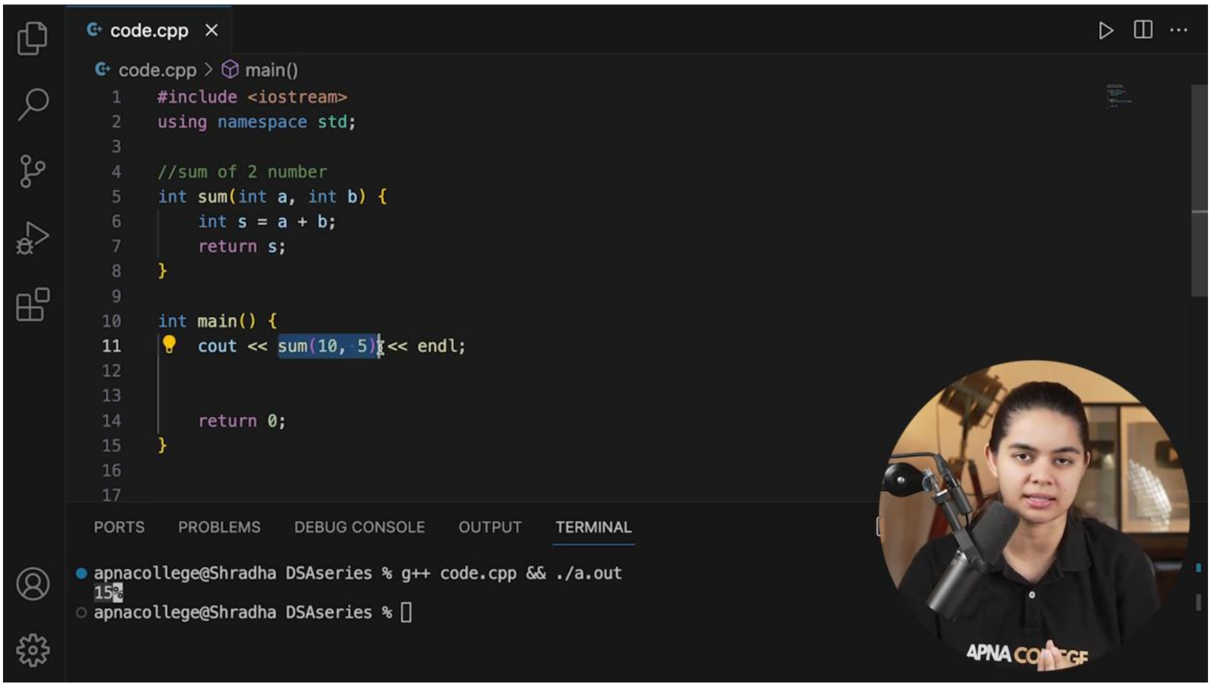
```
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
hello
val = 3
apnacollege@Shradha DSAseries % 
```



Function Syntax

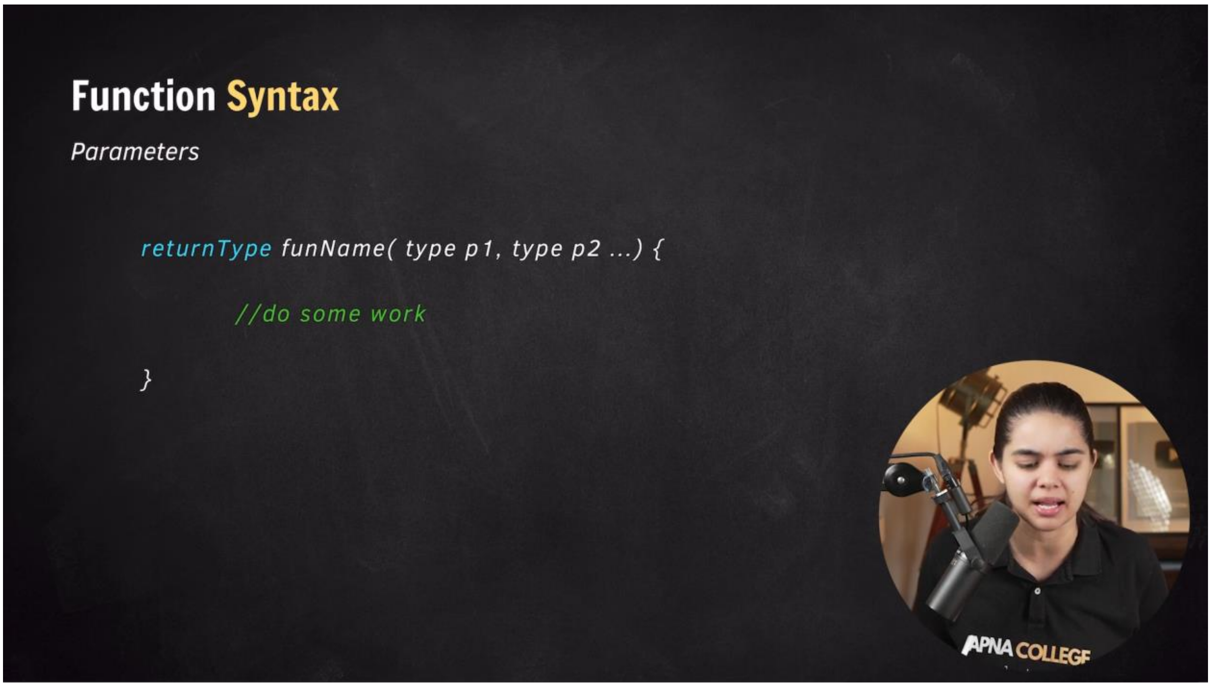
Parameters

```
returnType funName( type p1, type p2 ...) {

        //do some work

}
```



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

//sum of 2 number
int sum(int a, int b) {
    int s = a + b;
    return s;
}

int main() {
    cout << sum(10, 5) << endl;


    return 0;
}
```

```
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
15%
apnacollege@Shradha DSAseries % 
```

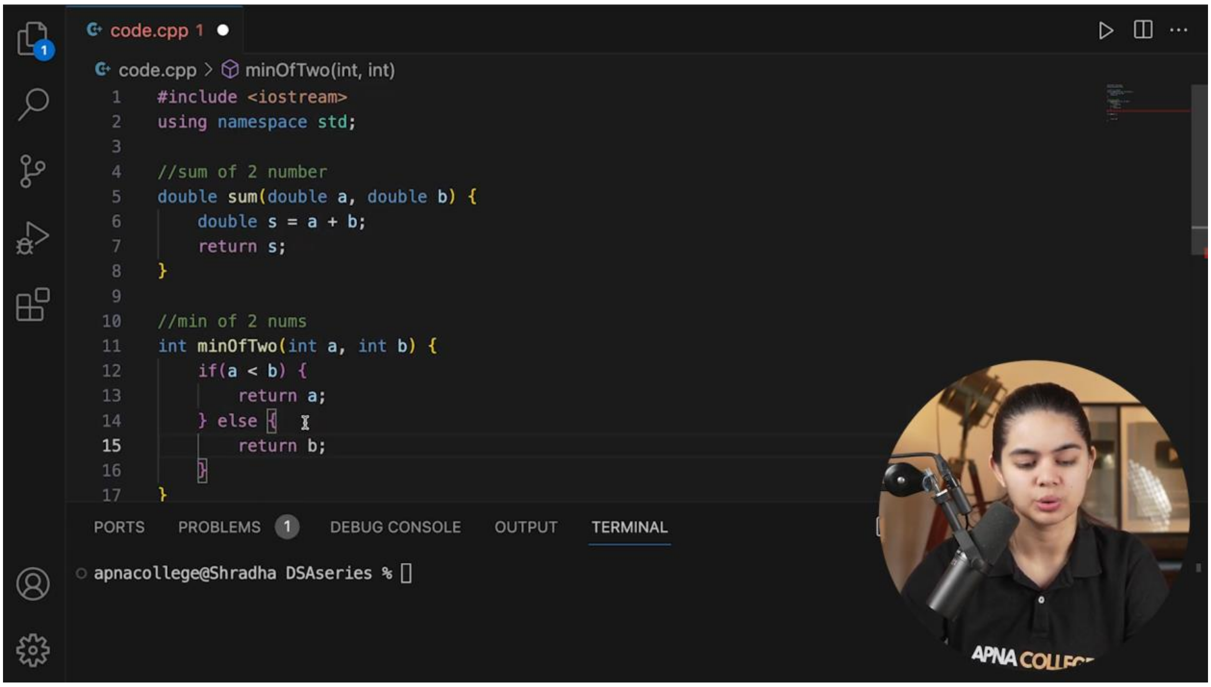


Function Syntax

Parameters

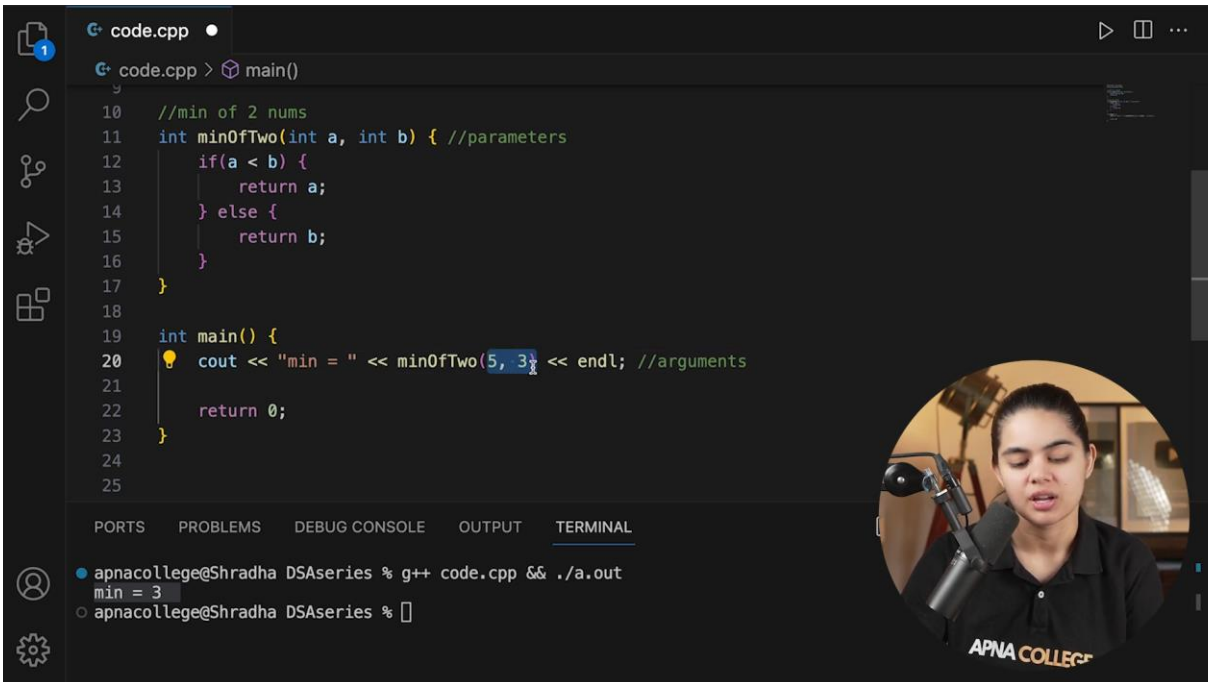
```
returnType funName( type p1, type p2 ...) {

        //do some work

}
```

APNA COLLEGE



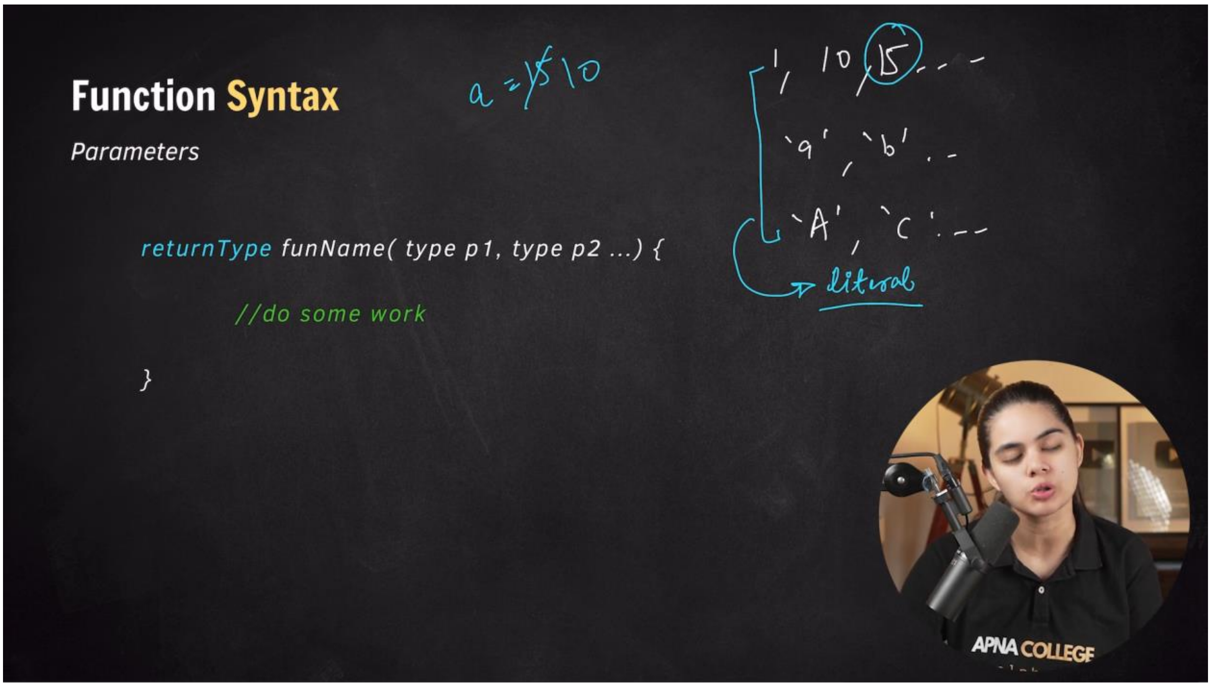
```cpp
#include <iostream>
using namespace std;

//sum of 2 number
double sum(double a, double b) {
    double s = a + b;
    return s;
}

//min of 2 nums
int minOfTwo(int a, int b) {
    if(a < b) {
        return a;
    } else {
        return b;
    }
}
```

```
apnacollege@Shradha DSAseries % 
```



```cpp
//min of 2 nums
int minOfTwo(int a, int b) { //parameters
    if(a < b) {
        return a;
    } else {
        return b;
    }
}

int main() {
    cout << "min = " << minOfTwo(5, 3) << endl; //arguments

    return 0;
}
```

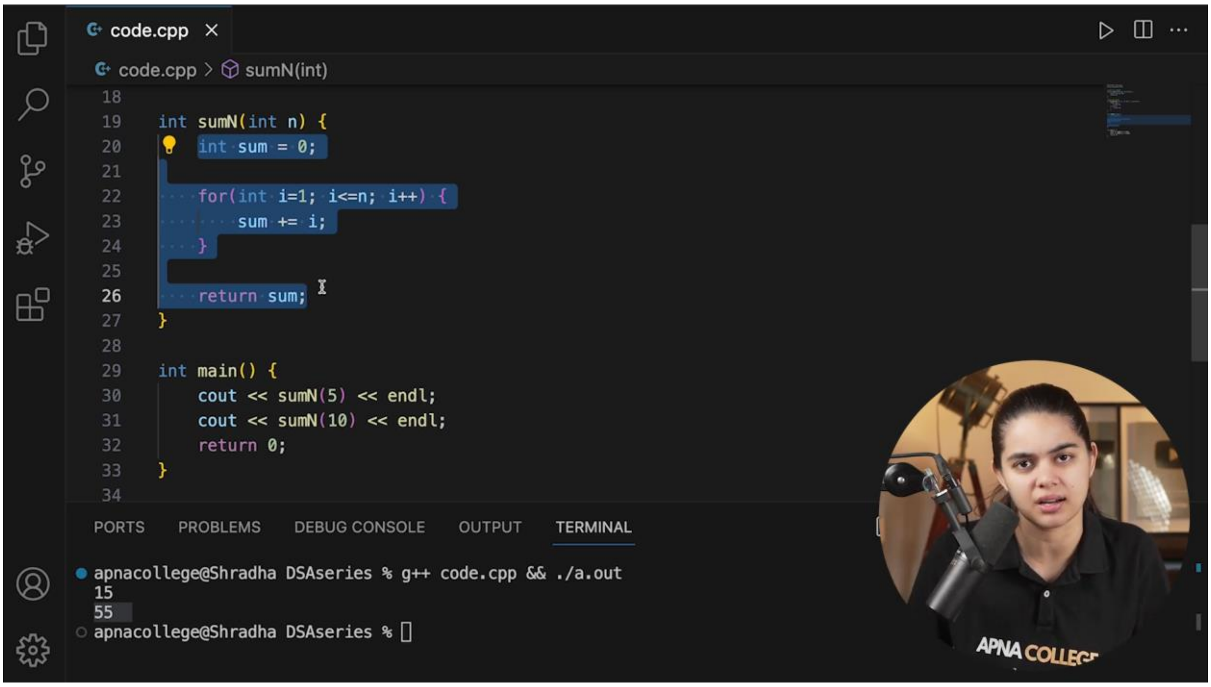
```
apnacollege@Shradha DSASeries % g++ code.cpp && ./a.out
min = 3
apnacollege@Shradha DSASeries % 
```



Function Syntax

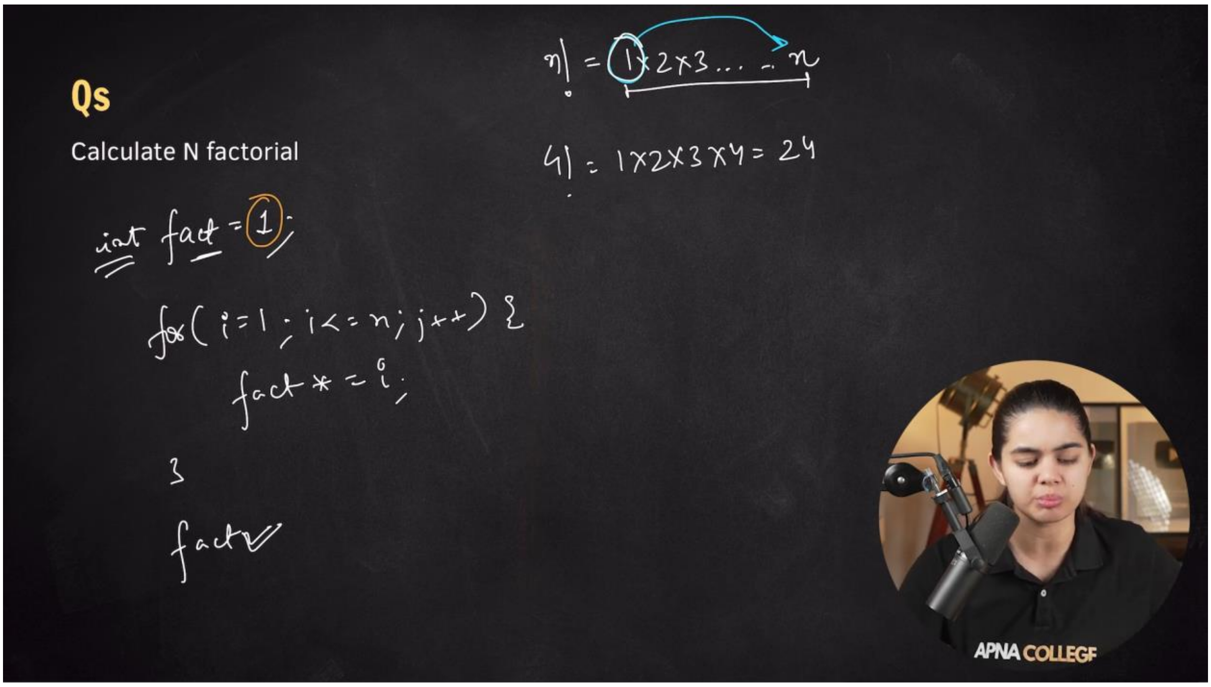
Parameters

```
returnType funName( type p1, type p2 ...) {

        //do some work

}
```



```cpp
int sumN(int n) {
    int sum = 0;

    for(int i=1; i<=n; i++) {
        sum += i;
    }

    return sum;
}

int main() {
    cout << sumN(5) << endl;
    cout << sumN(10) << endl;
    return 0;
}
```

```
apnacollege@Shradha DSASeries % g++ code.cpp && ./a.out
15
55
apnacollege@Shradha DSASeries % 
```



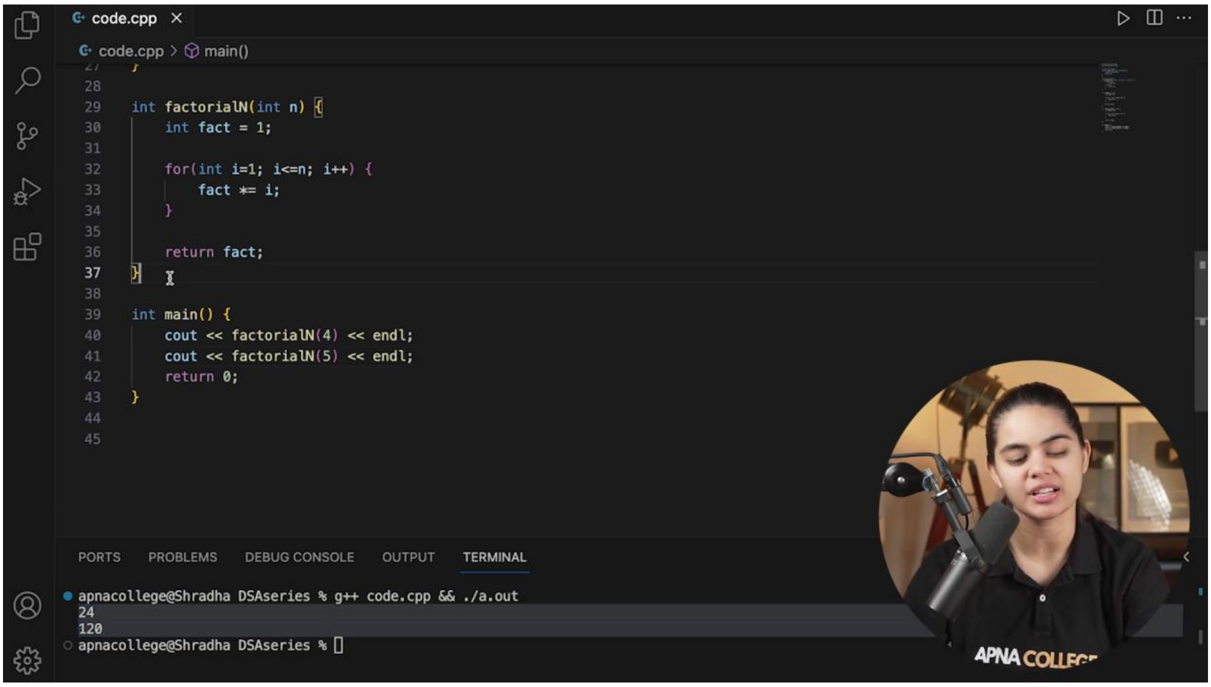
Qs

Calculate N factorial

$n! = 1 \times 2 \times 3 \ldots n$

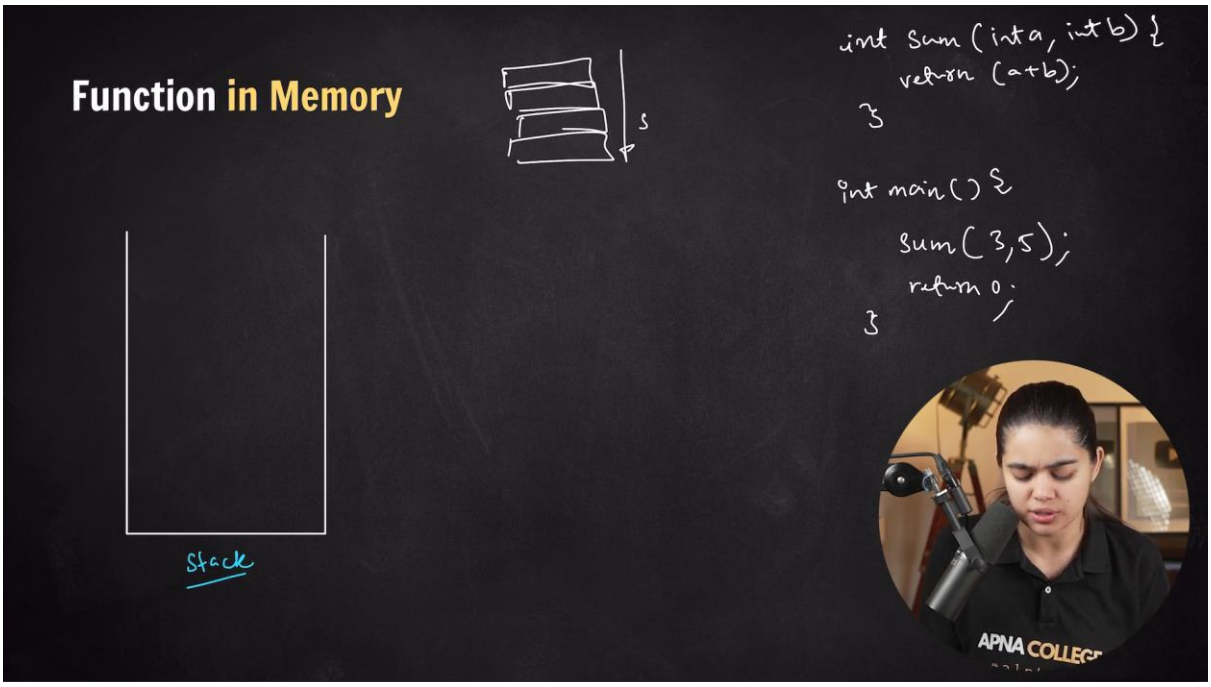
$4! = 1 \times 2 \times 3 \times 4 = 24$

```
int fact = 1;

for( i=1 ; i<=n; i++) {
        fact *= i;

}

fact
```



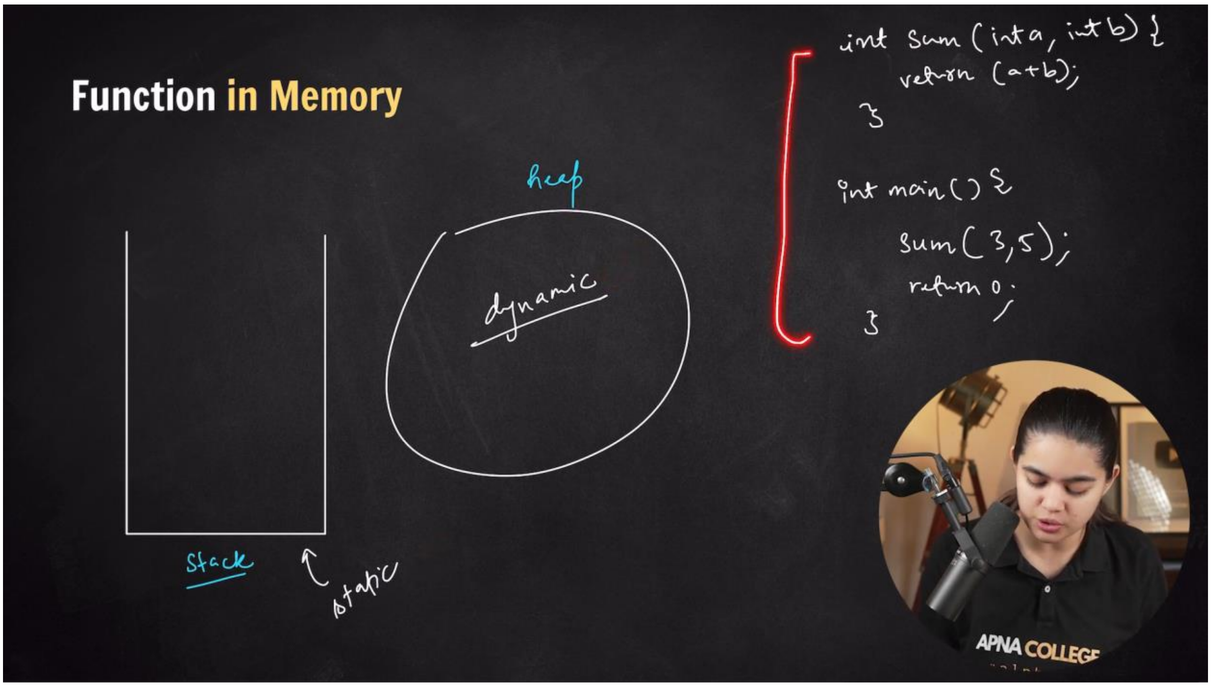
```cpp
int factorialN(int n) {
    int fact = 1;

    for(int i=1; i<=n; i++) {
        fact *= i;
    }

    return fact;
}

int main() {
    cout << factorialN(4) << endl;
    cout << factorialN(5) << endl;
    return 0;
}
```

```
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
24
120
apnacollege@Shradha DSAseries %
```

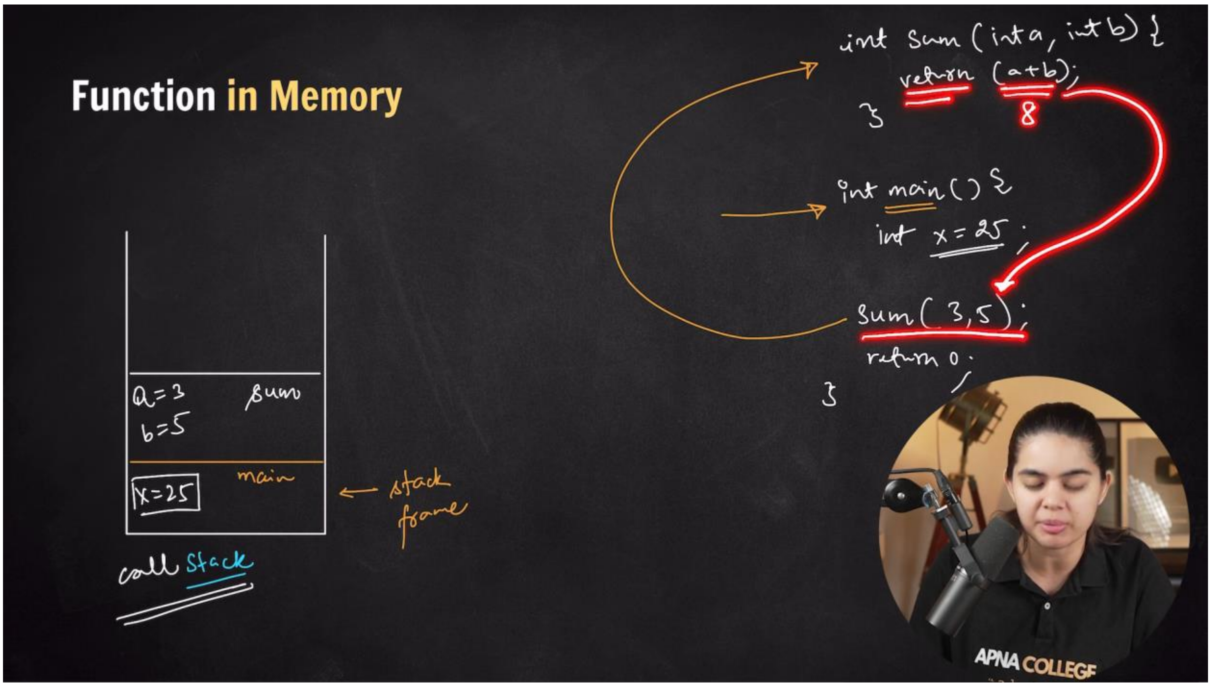


Function in Memory

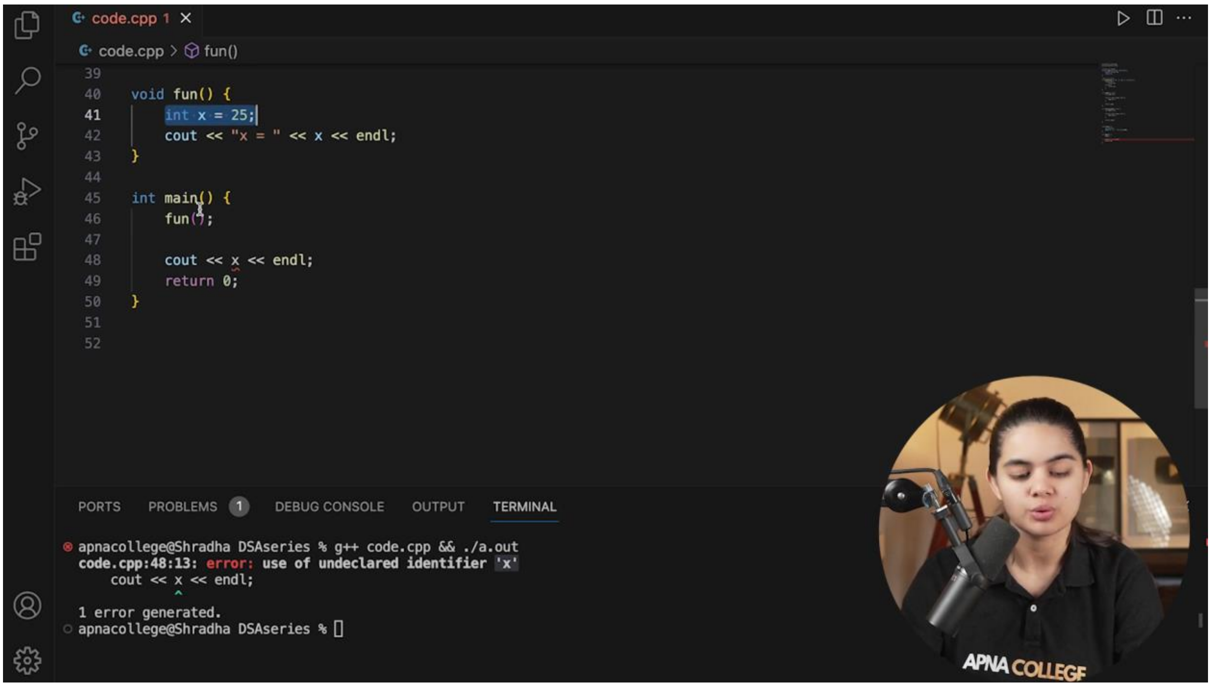
```
int sum (int a, int b) {
    return (a+b);
}

int main() {
    sum(3,5);
    return 0;
}
```



Function in Memory

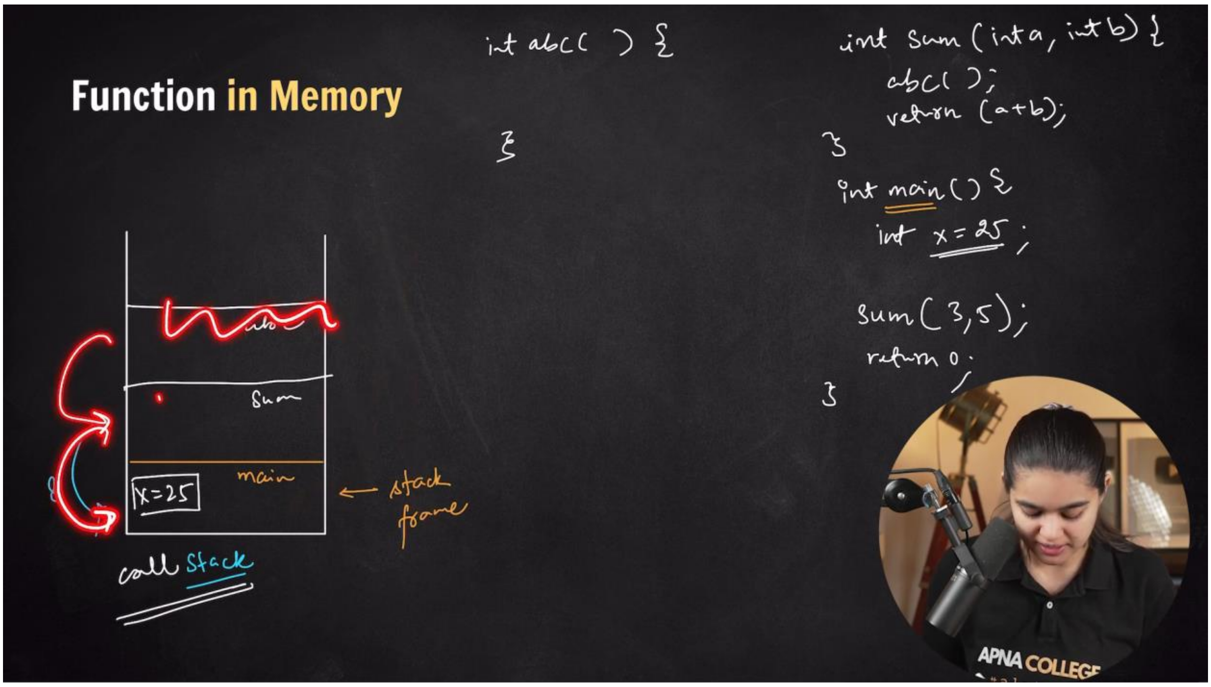


Function in Memory



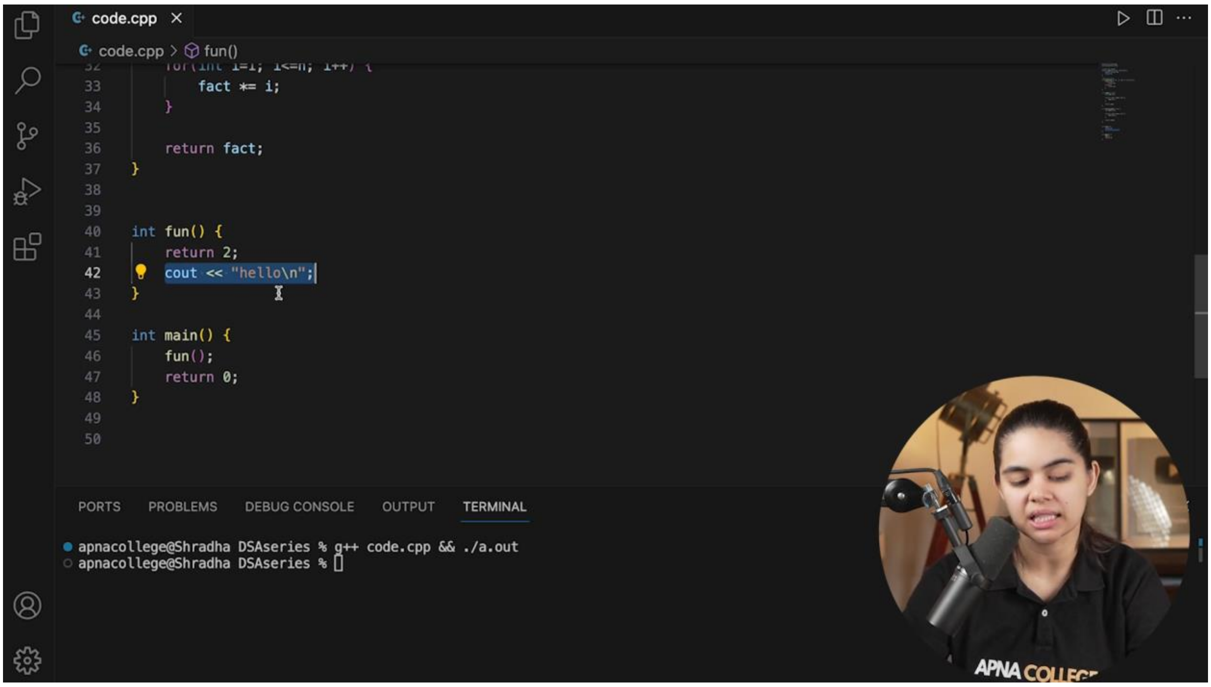
```cpp

void fun() {
    int x = 25;
    cout << "x = " << x << endl;
}

int main() {
    fun();

    cout << x << endl;
    return 0;
}
```

```
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
code.cpp:48:13: error: use of undeclared identifier 'x'
    cout << x << endl;
            ^
1 error generated.
apnacollege@Shradha DSAseries % 
```



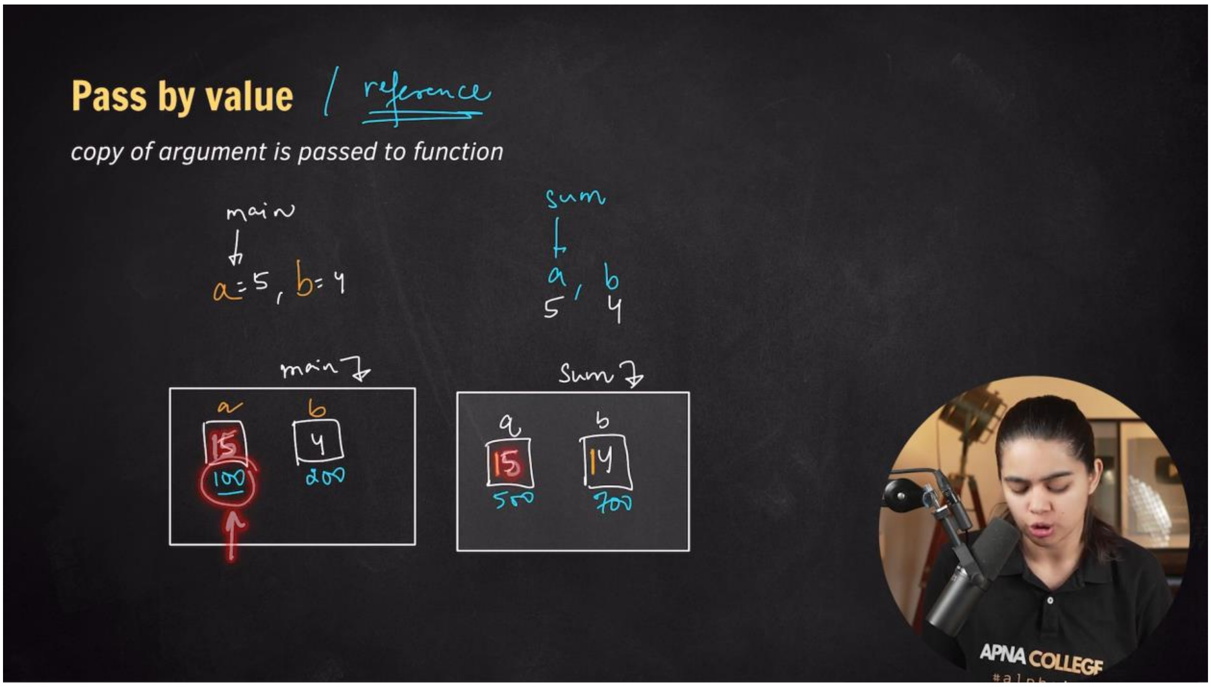
Function in Memory

```
int abc( ) {

}

int sum (int a, int b) {
    abc( );
    return (a+b);
}
int main( ) {
    int x = 25;

    sum(3,5);
    return 0;
}
```



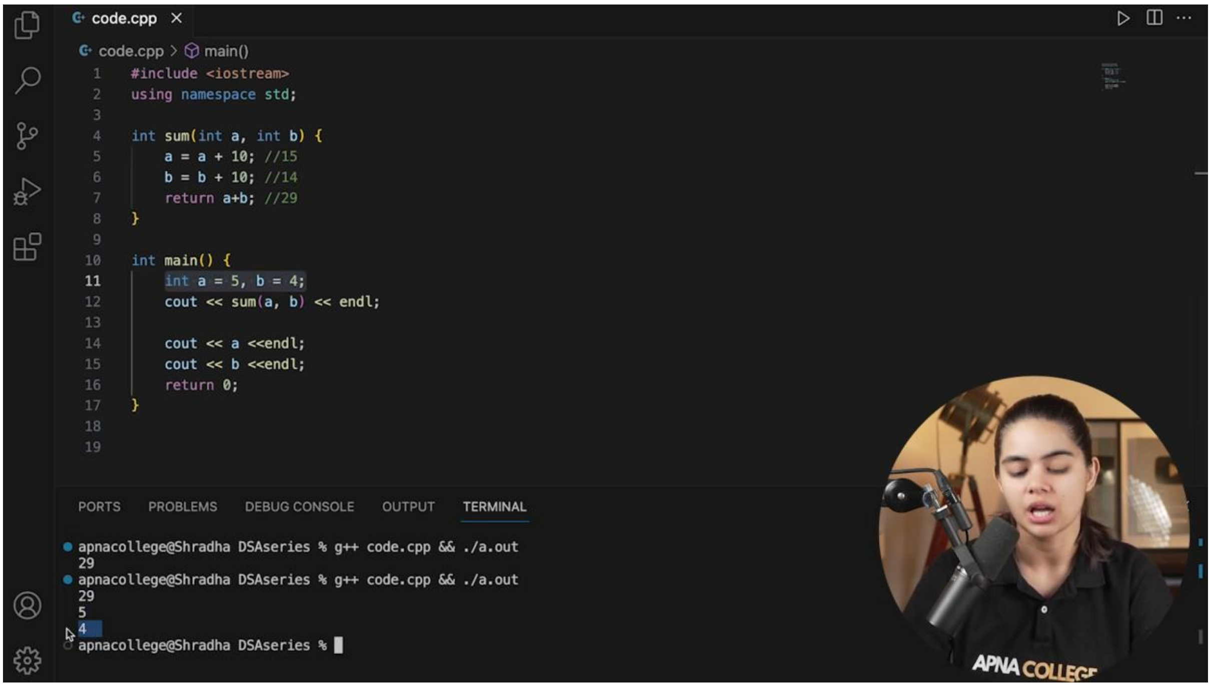
```
    for(int i=1; i<=n; i++) {
        fact *= i;
    }

    return fact;
}


int fun() {
    return 2;
    cout << "hello\n";
}

int main() {
    fun();
    return 0;
}
```

```
apnacollege@Shradha DSASeries % g++ code.cpp && ./a.out
apnacollege@Shradha DSASeries % 
```



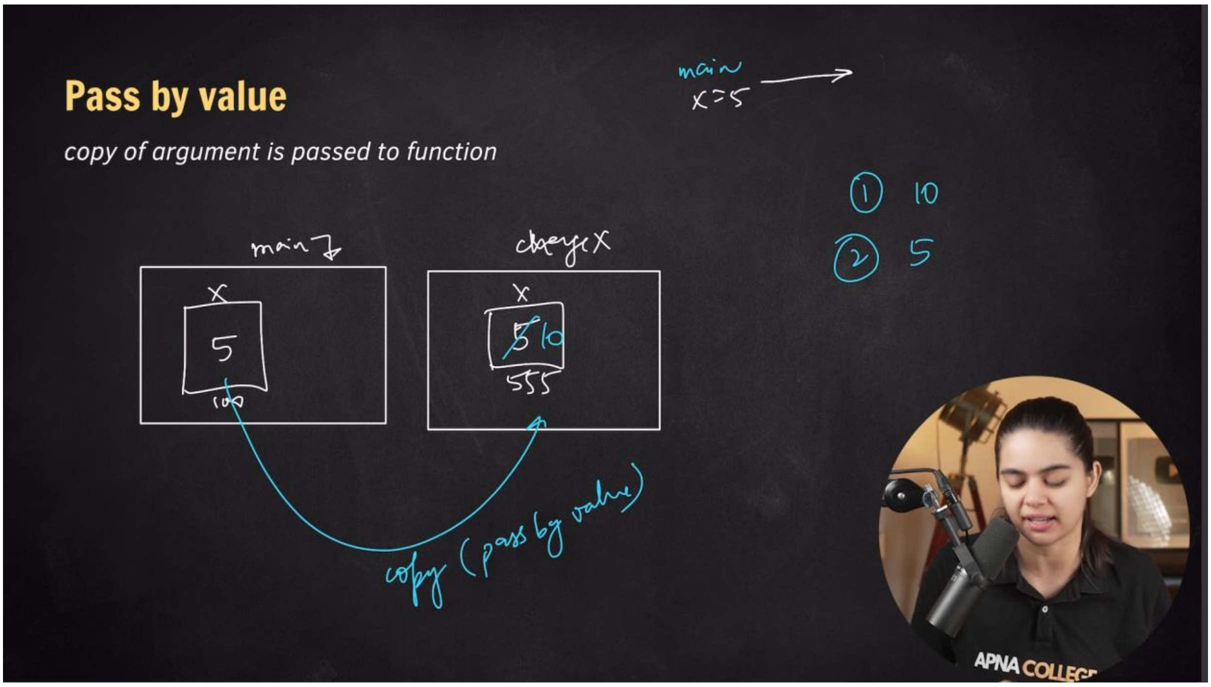
Pass by value / reference

copy of argument is passed to function



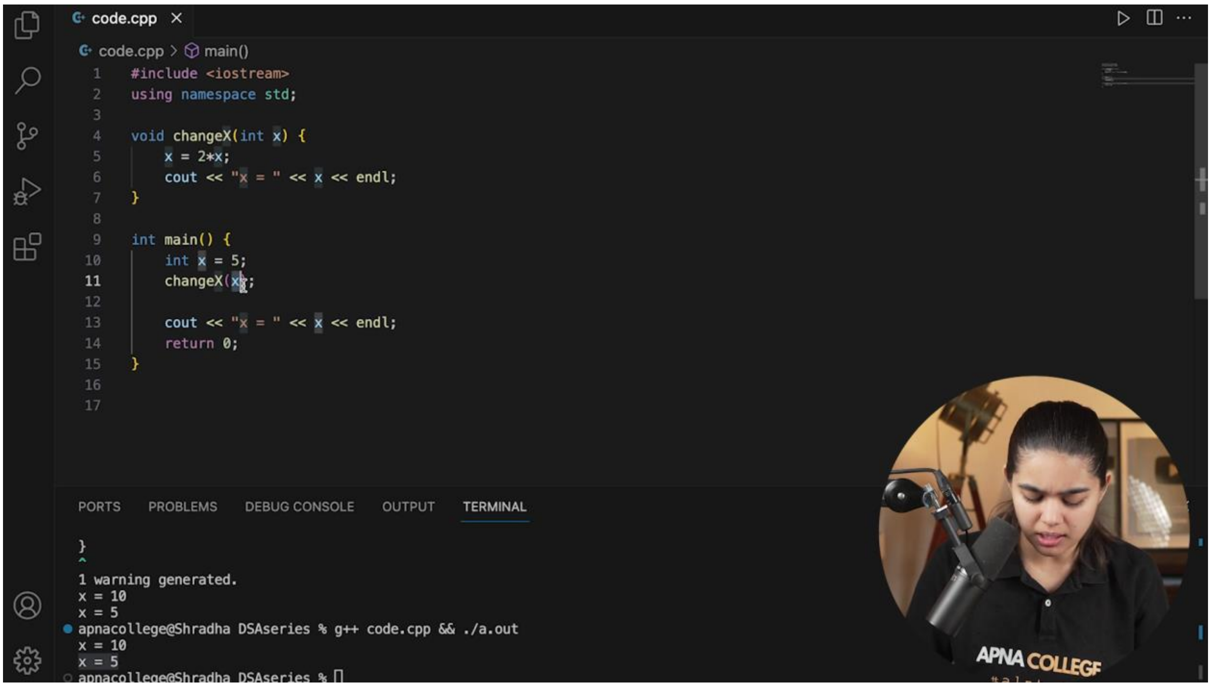
```cpp
#include <iostream>
using namespace std;

int sum(int a, int b) {
    a = a + 10; //15
    b = b + 10; //14
    return a+b; //29
}

int main() {
    int a = 5, b = 4;
    cout << sum(a, b) << endl;

    cout << a <<endl;
    cout << b <<endl;
    return 0;
}
```

```
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
29
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
29
5
4
apnacollege@Shradha DSAseries %
```



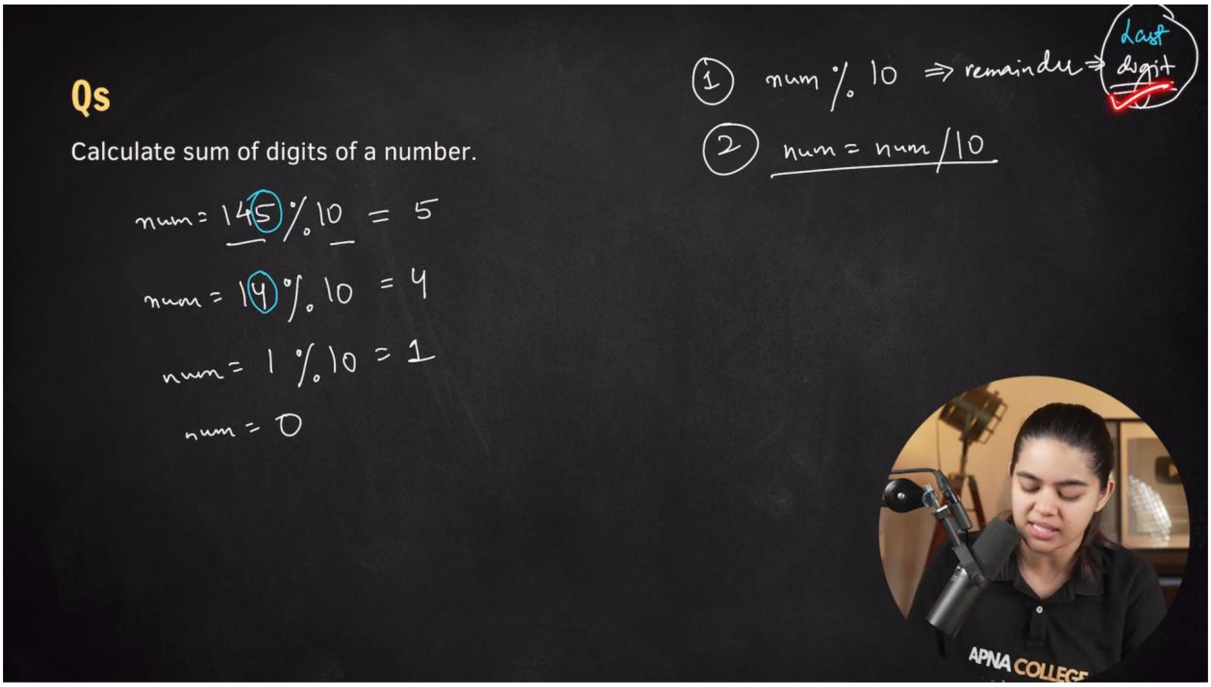
Pass by value

copy of argument is passed to function



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

void changeX(int x) {
    x = 2*x;
    cout << "x = " << x << endl;
}

int main() {
    int x = 5;
    changeX(x);

    cout << "x = " << x << endl;
    return 0;
}
```

```
}
^
1 warning generated.
x = 10
x = 5
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
x = 10
x = 5
apnacollege@Shradha DSAseries %
```



Qs

Calculate sum of digits of a number.

① num % 10 ⇒ remainder ⇒ last digit ✓

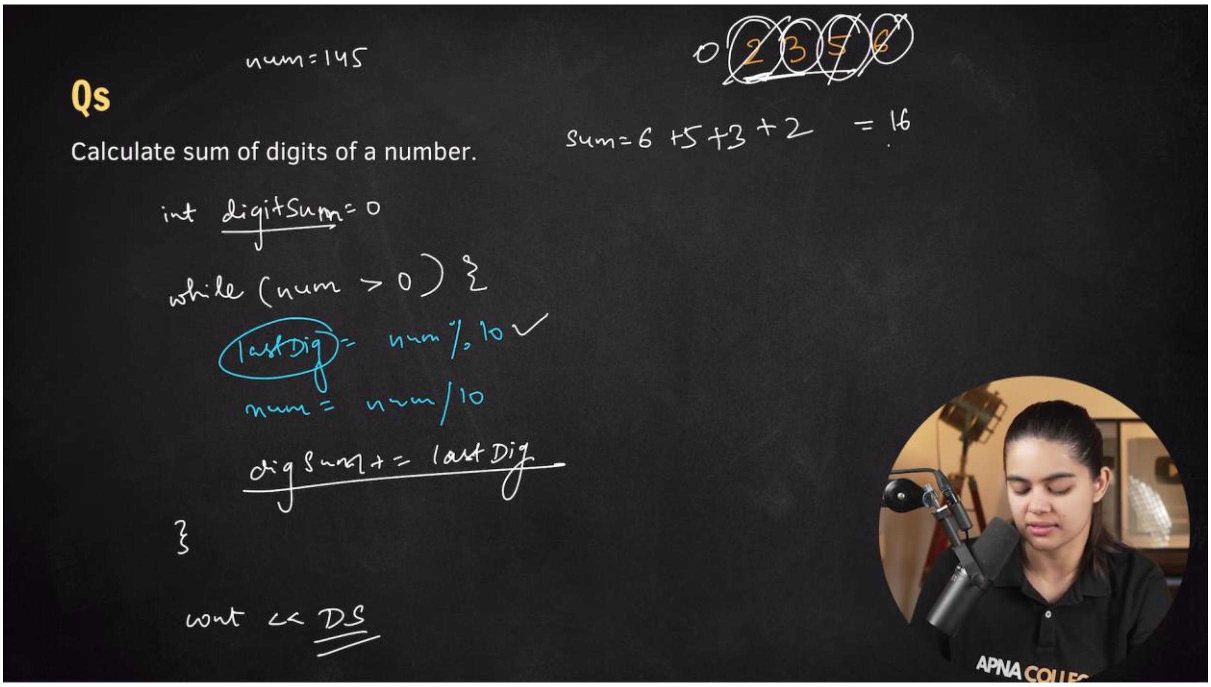
② num = num / 10

num = 145 % 10 = 5

num = 14 % 10 = 4

num = 1 % 10 = 1

num = 0



Qs

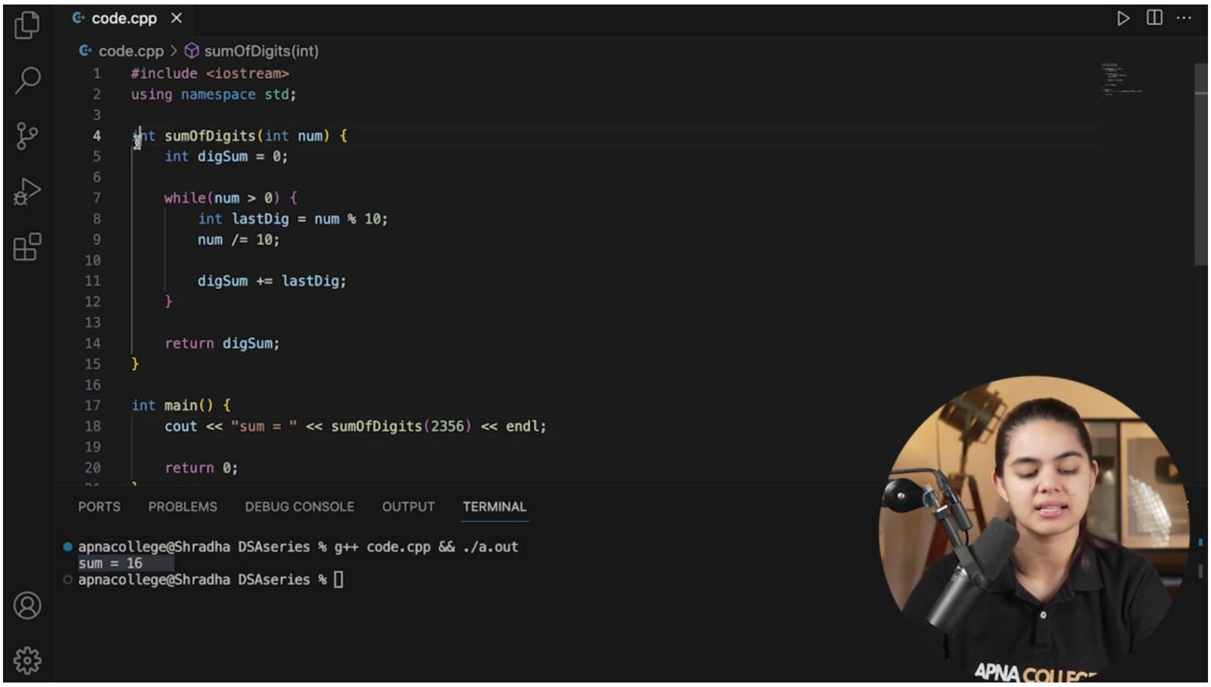
Calculate sum of digits of a number.

num = 145

~~0 2 3 5 6~~

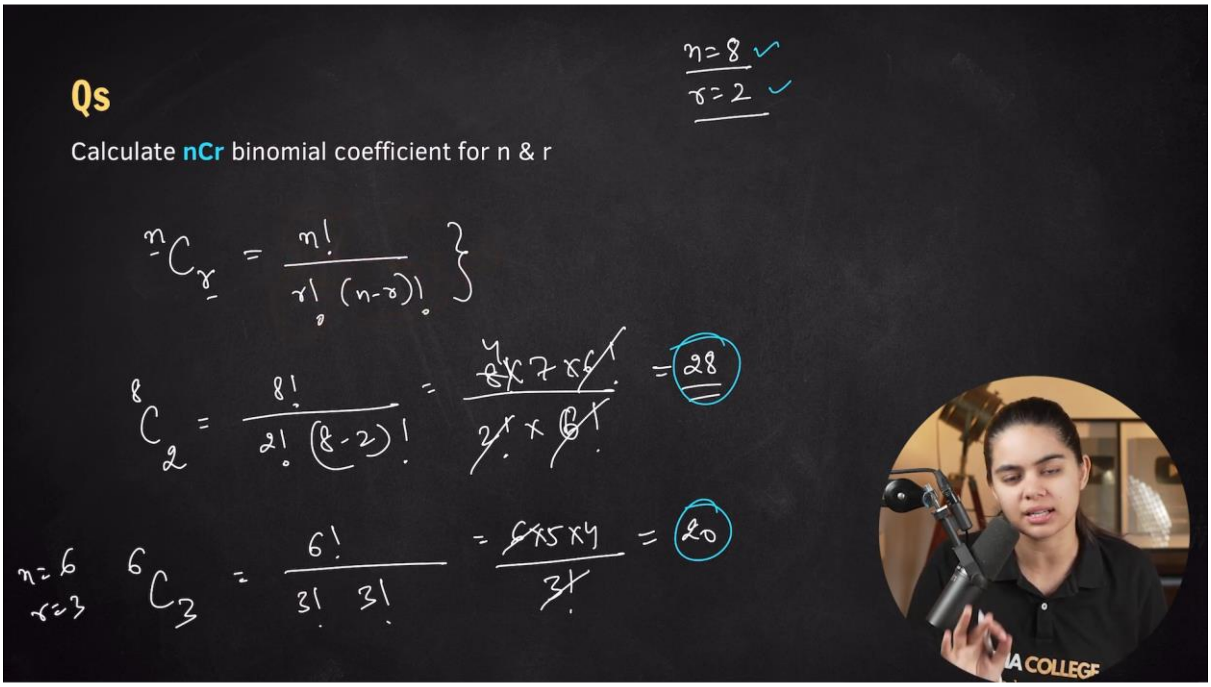
sum = 6 + 5 + 3 + 2 = 16

```
int digitSum = 0
while (num > 0) {
    lastDig = num % 10 ✓
    num = num / 10
    digSum += lastDig
}
cout << DS
```



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

int sumOfDigits(int num) {
    int digSum = 0;

    while(num > 0) {
        int lastDig = num % 10;
        num /= 10;

        digSum += lastDig;
    }

    return digSum;
}

int main() {
    cout << "sum = " << sumOfDigits(2356) << endl;

    return 0;
```

```
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
sum = 16
apnacollege@Shradha DSAseries % 
```



Qs

Calculate **nCr** binomial coefficient for n & r

$n=8$ ✓

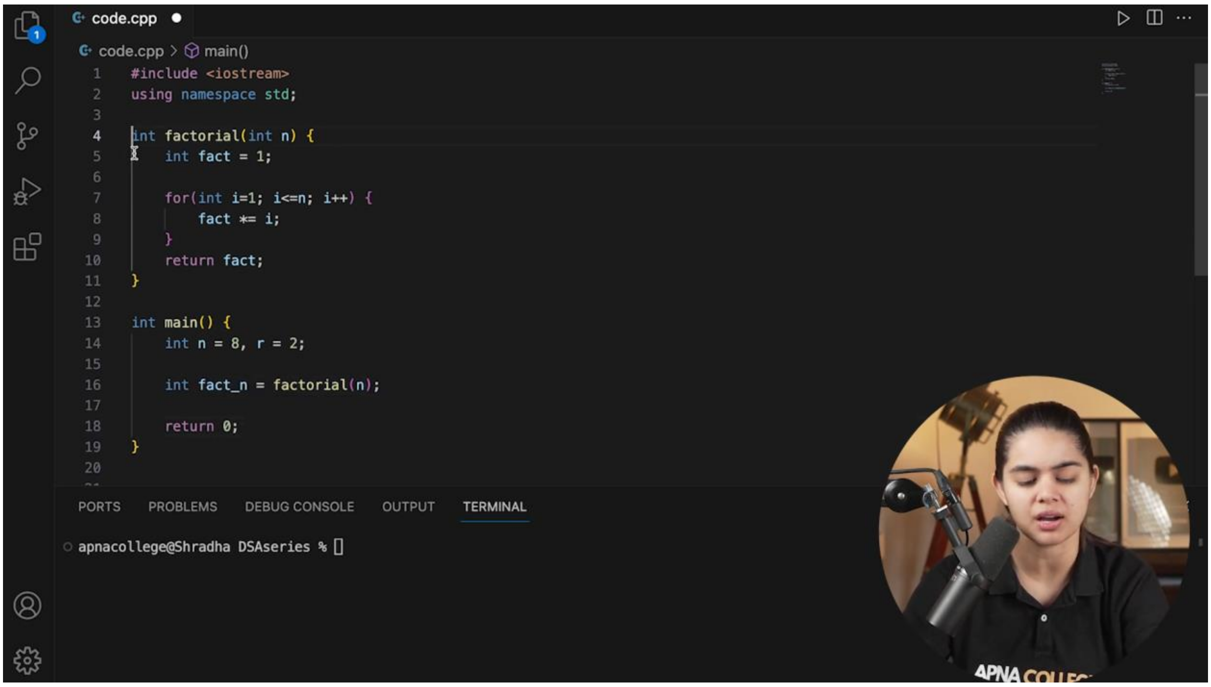
$r=2$ ✓

$${}^{n}C_{r} = \frac{n!}{r!\,(n-r)!}$$

$${}^{8}C_{2} = \frac{8!}{2!\,(8-2)!} = \frac{\cancel{8}^{4} \times 7 \times \cancel{6!}}{\cancel{2!} \times \cancel{6!}} = \boxed{28}$$

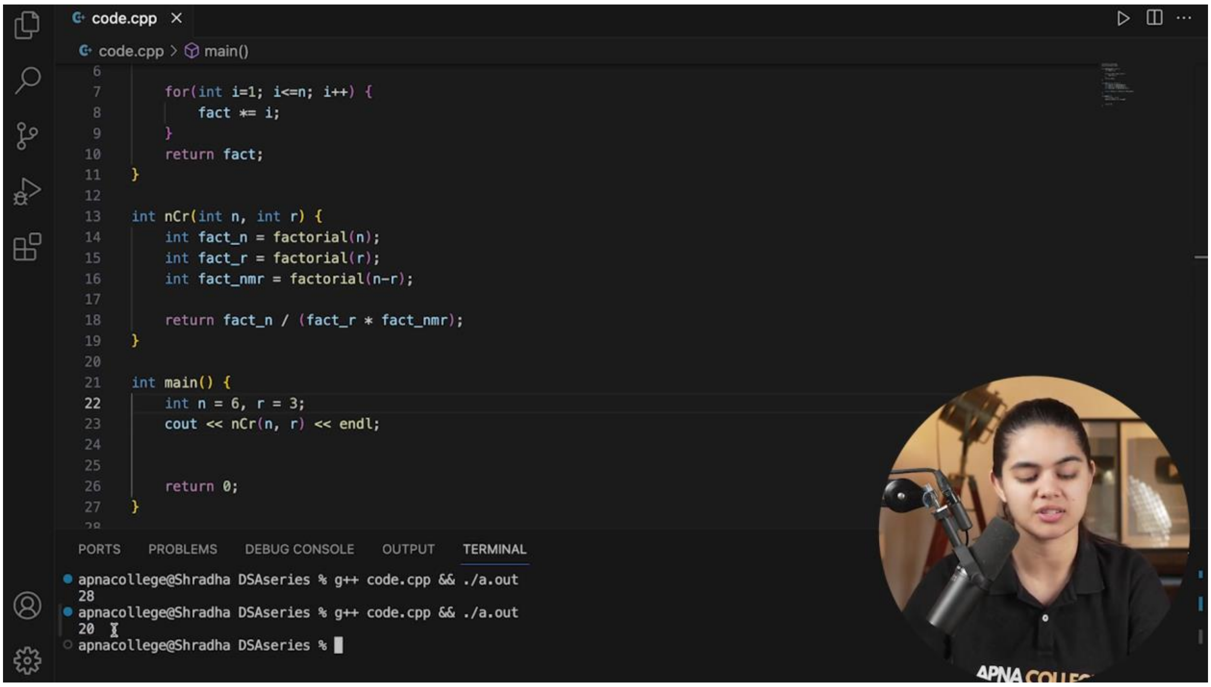
$n=6$
$r=3$

$${}^{6}C_{3} = \frac{6!}{3!\;3!} = \frac{\cancel{6}\times 5 \times 4}{\cancel{3!}} = \boxed{20}$$



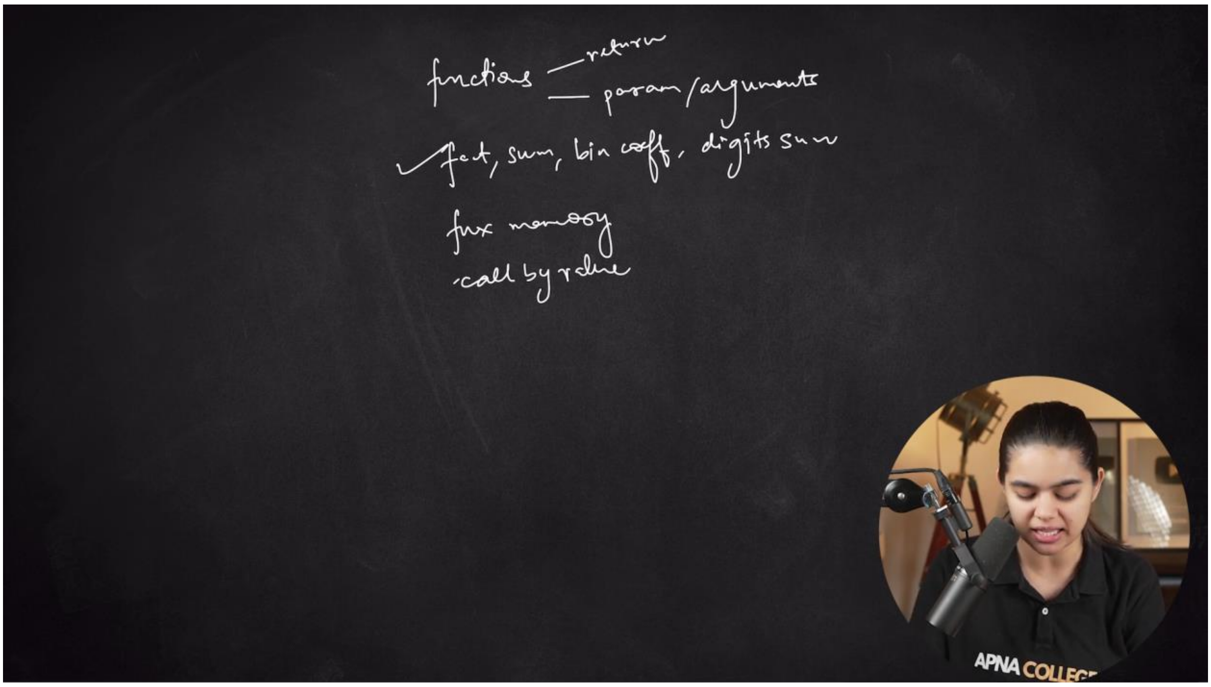
```cpp
#include <iostream>
using namespace std;

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

    for(int i=1; i<=n; i++) {
        fact *= i;
    }
    return fact;
}

int main() {
    int n = 8, r = 2;

    int fact_n = factorial(n);

    return 0;
}
```

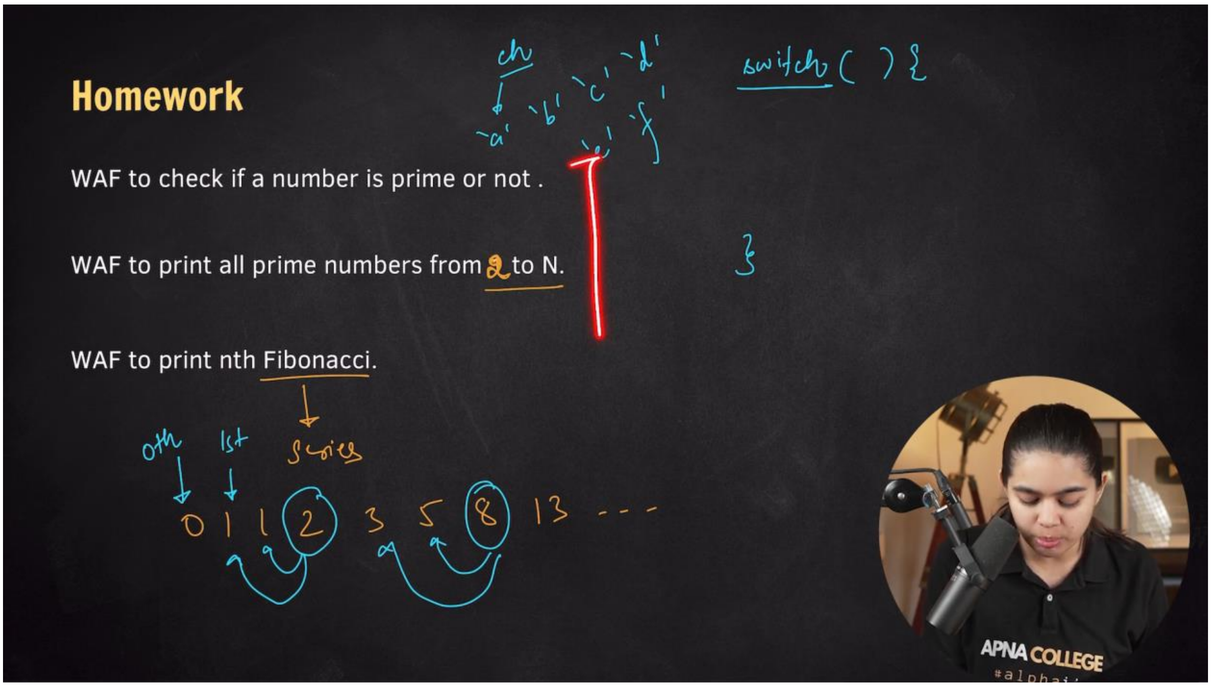
```
apnacollege@Shradha DSAseries %
```



```cpp

    for(int i=1; i<=n; i++) {
        fact *= i;
    }
    return fact;
}

int nCr(int n, int r) {
    int fact_n = factorial(n);
    int fact_r = factorial(r);
    int fact_nmr = factorial(n-r);

    return fact_n / (fact_r * fact_nmr);
}

int main() {
    int n = 6, r = 3;
    cout << nCr(n, r) << endl;


    return 0;
}
```

```
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
28
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
20
apnacollege@Shradha DSAseries %
```





Homework

WAF to check if a number is prime or not .

WAF to print all prime numbers from 2 to N.

WAF to print nth Fibonacci.

Homework Solution:

```
// Write a function to check if a number is prime or not .


#include <iostream>

using namespace std;


int checkPrime(int num) {

   bool isPrime = true;


   if(num <= 2 && num > 0) {
     cout << "Number is prime" << endl;
     return 0;
   } else if(num <= 0) {
     cout << "Number is too small!" << endl;
     return 0;
   }
   for(int i = 2; i < num; i++) {
     if(num%i == 0) {
       isPrime = false;
       break;
     }
   }

   if(isPrime == true) {
     cout << "Number is prime" << endl;
   } else {
     cout << "Number is not prime" << endl;
   }

   return 0;
}

int main() {
   int n = 18;

   checkPrime(n);

   return 0;
}
```

```
// Write a function to print all prime numbers from 2 to N.

#include <iostream>
using namespace std;

void checkPrime(int num) {
    bool isPrime = true;

    if (num <= 1) {
        // Numbers less than or equal to 1 are not prime
        return;
    }

    for (int i = 2; i * i <= num; i++) {
        if (num % i == 0) {
            isPrime = false;
            break;
        }
    }

    if (isPrime) {
        cout << num << " ";
    }
}

void printPrime(int nn) {
    for (int i = 2; i <= nn; i++) {
        checkPrime(i);
    }
    cout << endl;
}

int main() {
    int n = 20;
    printPrime(n);
```

```
  return 0;
}

// #include <iostream>
// using namespace std;

// void checkPrime(int num) {
//    bool isPrime = true;

//    if(num <= 2 && num > 0) {
//       cout << "Number is prime" << endl;
//       return;
//    } else if(num <= 0) {
//       cout << "Number is too small!" << endl;
//       return;
//    }
//    for(int i = 2; i < num; i++) {
//       if(num%i == 0) {
//          isPrime = false;
//          break;
//       }
//    }

//    if(isPrime == true) {
//       cout << num << endl;
//    }

//    return;
// }

// char printPrime(int nn) {

//    for(int i = 2; i <= nn; i++) {
//       cout << checkPrime(i) << endl;
//    }
//    return;
// }

// int main() {
//    int n = 20;

//    printPrime(n);
//    return 0;
// }
```

```
// Write a function to print nth Fibonacci

#include <iostream>
using namespace std;

void printFibonacci(int num) {
  int i = 0, j = 1;
  int nextElement = 0;

  while(nextElement <= num) {
    nextElement = i + j;

    if(i == 0) {
      cout << i << " ";
    } else if(i == 1) {
      cout << i << " ";
    }

    i = j;
    j = nextElement;

    if(nextElement <= num) {
            cout << nextElement << " ";
        }

    }
    return;
}

int main() {
    int num = 20;
    printFibonacci(num);
    return 0;
}

// #include <iostream>
// using namespace std;

// int main() {
//     int n, t1 = 0, t2 = 1, nextTerm = 0;

//     cout << "Enter the number of terms: ";
//     cin >> n;

//     cout << "Fibonacci Series: ";

//     for (int i = 1; i <= n; ++i) {
//         // Prints the first two terms.
//         if(i == 1) {
//             cout << t1 << " ";
//             continue;
//         }
//         if(i == 2) {
//             cout << t2 << " ";
//             continue;
//         }
//         nextTerm = t1 + t2;
//        t1 = t2;

//        t2 = nextTerm;


//        cout << nextTerm << " ";

//    }

//    return 0;

// }
```