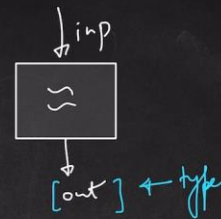


Chapter-5 Functions

Function Syntax

```
int main() {  
    return 0;  
}  
  
returnType printHello() {  
    cout << "hello";  
}
```



```
code.cpp  
1 #include <iostream>  
2 using namespace std;  
3  
4 //function definition  
5 void printHello() {  
6     cout << "hello\n";  
7 }  
8  
9 int main() {  
10     printHello();  
11  
12     int a = 10;  
13     int b = 5;  
14  
15     cout << "hello\n";  
16     return 0;  
17 }
```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT TERMINAL

apnacollege@Shradha DSASeries %

```
code.cpp x
code.cpp > main()
//function definition
5 void printHello() {
6     cout << "hello\n";
7 }
8
9 int main() {
10     //function call / invoke
11     printHello();
12     printHello();
13     printHello();
14     printHello();
15     printHello();
16
17     return 0;
18 }
19
20
```

hello
hello
hello
hello
hello

apnacollege@Shradha DSAseries %

```
code.cpp x
code.cpp > main()
//function definition
2 using namespace std;
3
4 //function definition
5 int printHello() {
6     cout << "hello\n";
7     return 3;
8 }
9
10 int main() {
11     //function call / invoke
12     int val = printHello();
13     cout << "val = " << val << endl;
14
15     return 0;
16 }
17
```

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

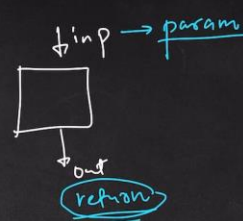
hello
val = 3

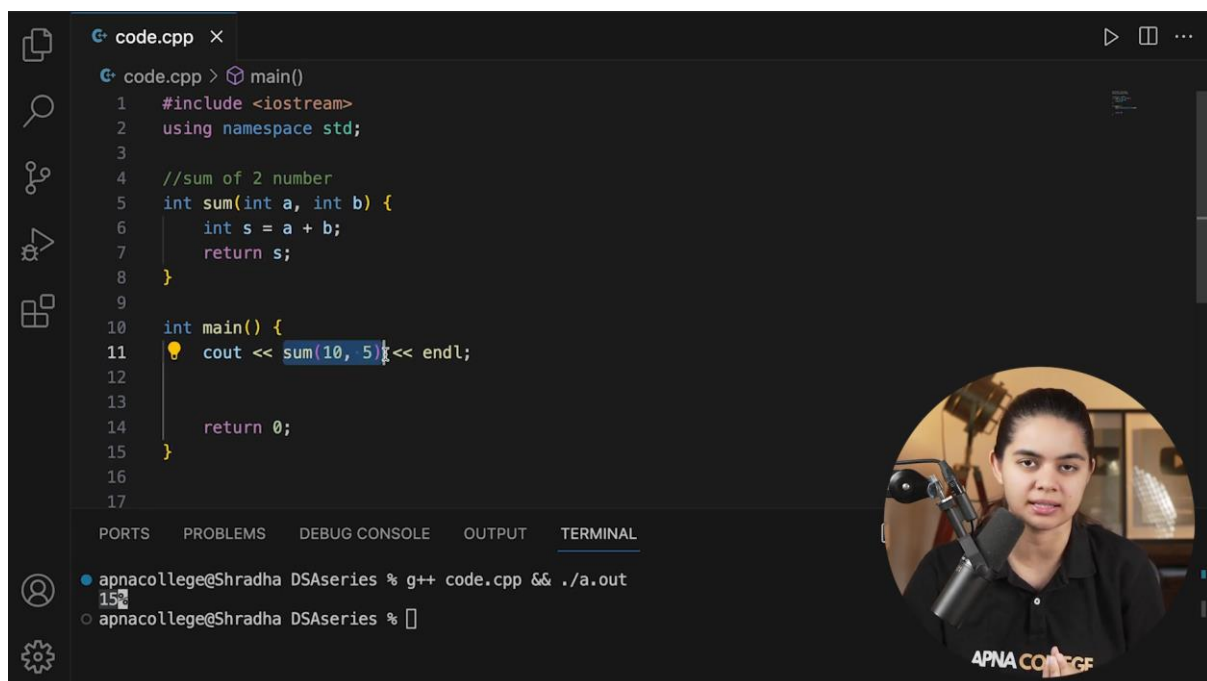
apnacollege@Shradha DSAseries %

Function Syntax

Parameters

```
returnType funName( type p1, type p2 ... ) {
    //do some work
}
```






```
code.cpp x
code.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  //sum of 2 number
5  int sum(int a, int b) {
6      int s = a + b;
7      return s;
8  }
9
10 int main() {
11     cout << sum(10, 5) << endl;
12
13
14     return 0;
15 }
16
17
```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT TERMINAL

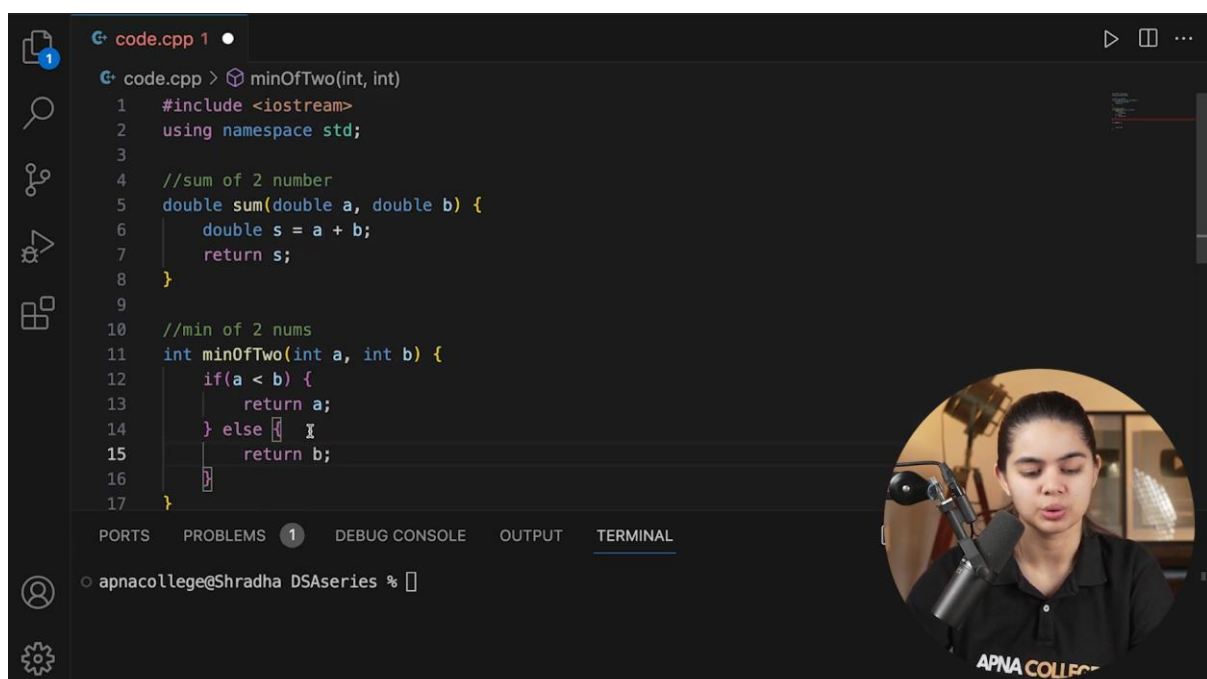
apnacollege@Shradha DSASeries % g++ code.cpp && ./a.out
15
apnacollege@Shradha DSASeries %



Function Syntax

Parameters

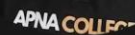
```
returnType funName( type p1, type p2 ...) {  
  
    //do some work  
  
}
```



```
code.cpp 1
code.cpp > minOfTwo(int, int)
1  #include <iostream>
2  using namespace std;
3
4  //sum of 2 number
5  double sum(double a, double b) {
6      double s = a + b;
7      return s;
8  }
9
10 //min of 2 nums
11 int minOfTwo(int a, int b) {
12     if(a < b) {
13         return a;
14     } else {
15         return b;
16     }
17 }
```

PORTS PROBLEMS 1 DEBUG CONSOLE OUTPUT TERMINAL


apnacollege@Shradha DSASeries %




```
code.cpp
code.cpp > main()
10 //min of 2 nums
11 int minOfTwo(int a, int b) { //parameters
12     if(a < b) {
13         return a;
14     } else {
15         return b;
16     }
17 }
18
19 int main() {
20     cout << "min = " << minOfTwo(5, 3) << endl; //arguments
21
22     return 0;
23 }
24
25
```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT TERMINAL

```
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
}
```

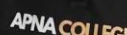
$a = 10$

1, 10, 15 ---
'a', 'b' ---
'A', 'C' ---
Literal

```
code.cpp
code.cpp > sumN(int)
18
19 int sumN(int n) {
20     int sum = 0;
21
22     for(int i=1; i<=n; i++) {
23         sum += i;
24     }
25
26     return sum;
27 }
28
29 int main() {
30     cout << sumN(5) << endl;
31     cout << sumN(10) << endl;
32     return 0;
33 }
34
```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT TERMINAL

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



Qs

Calculate N factorial

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

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

int fact = 1;

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

}

fact



```
code.cpp x
code.cpp > main()
28
29 int factorialN(int n) {
30     int fact = 1;
31
32     for(int i=1; i<=n; i++) {
33         fact *= i;
34     }
35
36     return fact;
37 }
38
39 int main() {
40     cout << factorialN(4) << endl;
41     cout << factorialN(5) << endl;
42     return 0;
43 }
44
45
```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT TERMINAL

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



Function in Memory



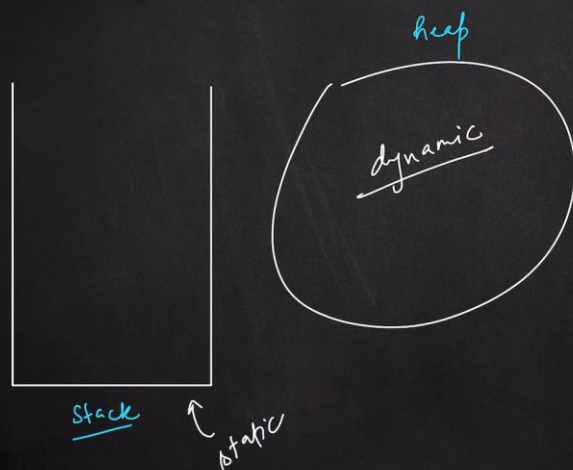
Stack

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

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



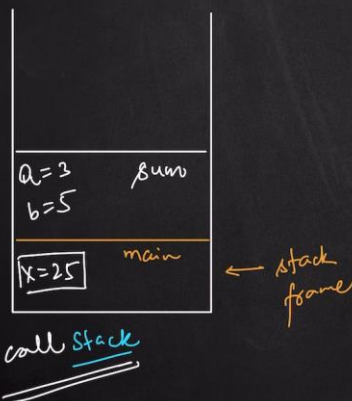
Function in Memory



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



Function in Memory



```
int sum(int a, int b) {  
    return (a+b);  
}  
  
int main() {  
    int x=25;  
    sum(3,5);  
    return 0;  
}
```



The screenshot shows a C++ code editor with the following code:

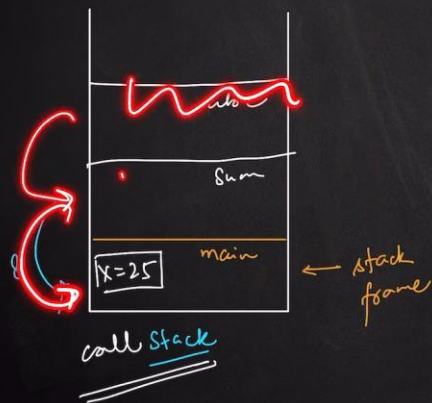
```
code.cpp 1 x  
code.cpp > fun()  
39  
40 void fun() {  
41     int x = 25;  
42     cout << "x = " << x << endl;  
43 }  
44  
45 int main() {  
46     fun();  
47  
48     cout << x << endl;  
49     return 0;  
50 }  
51  
52
```

The terminal output shows the following error:

```
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;
}
```



```
code.cpp x
code.cpp > fun()
32 fun(int i=1, i<=10; i++) {
33     fact *= i;
34 }
35
36 return fact;
37 }
38
39
40 int fun() {
41     return 2;
42     cout << "hello\n";
43 }
44
45 int main() {
46     fun();
47     return 0;
48 }
49
50
```

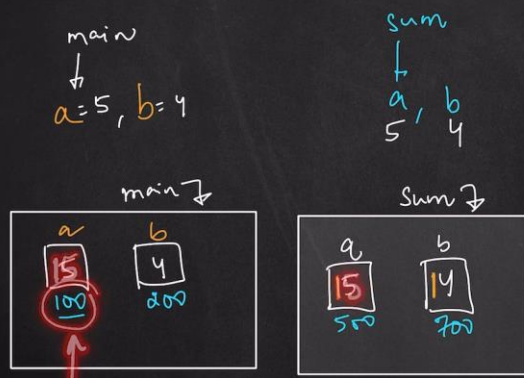
PORTS PROBLEMS DEBUG CONSOLE OUTPUT **TERMINAL**

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



Pass by value / reference

copy of argument is passed to function



```

code.cpp x
code.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int sum(int a, int b) {
5      a = a + 10; //15
6      b = b + 10; //14
7      return a+b; //29
8  }
9
10 int main() {
11     int a = 5, b = 4;
12     cout << sum(a, b) << endl;
13
14     cout << a << endl;
15     cout << b << endl;
16     return 0;
17 }
18
19


```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT **TERMINAL**

```

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

main

X
5

changeX


X
5

copy (pass by value)

main
x=5 →

① 10

② 5



```

code.cpp x
code.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  void changeX(int x) {
5      x = 2*x;
6      cout << "x = " << x << endl;
7  }
8
9  int main() {
10     int x = 5;
11     changeX(x);
12
13     cout << "x = " << x << endl;
14     return 0;
15 }
16
17


```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT **TERMINAL**

```

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.

$$\text{num} = 145 \% 10 = 5$$

$$\text{num} = 14 \% 10 = 4$$

$$\text{num} = 1 \% 10 = 1$$

$$\text{num} = 0$$

- ① $\text{num} \% 10 \Rightarrow \text{remainder} \Rightarrow \text{last digit}$
- ② $\text{num} = \text{num} / 10$



Qs

Calculate sum of digits of a number.

$\text{num} = 145$

0 2 3 5 6

$$\text{sum} = 6 + 5 + 3 + 2 = 16$$

$\text{int digSum} = 0$

$\text{while} (\text{num} > 0) \{$

$\text{lastDig} = \text{num} \% 10$ ✓

$\text{num} = \text{num} / 10$

$\text{digSum} += \text{lastDig}$

$\}$

$\text{cout} << \underline{\underline{\text{DS}}}$



```
code.cpp x
code.cpp > sumOfDigits(int)
1 #include <iostream>
2 using namespace std;
3
4 int sumOfDigits(int num) {
5     int digSum = 0;
6
7     while(num > 0) {
8         int lastDig = num % 10;
9         num /= 10;
10
11         digSum += lastDig;
12     }
13
14     return digSum;
15 }
16
17 int main() {
18     cout << "sum = " << sumOfDigits(2356) << endl;
19
20     return 0;
21 }
```

PORTS PROBLEMS DEBUG CONSOLE OUTPUT TERMINAL

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

apnacollege@Shradha DSAseries %



Qs

Calculate nCr binomial coefficient for n & r

$$\begin{array}{r} n=8 \\ r=2 \\ \hline \end{array}$$

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

$${}^8C_2 = \frac{8!}{2! (8-2)!} = \frac{8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1}{2 \times 1 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1} = 28$$

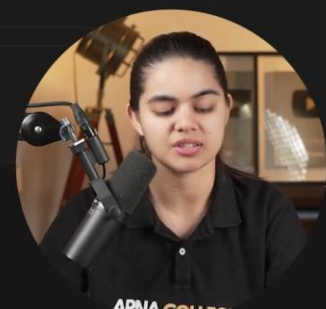
$${}^6C_3 = \frac{6!}{3! 3!} = \frac{6 \times 5 \times 4 \times 3 \times 2 \times 1}{3 \times 2 \times 1 \times 3 \times 2 \times 1} = 20$$



```
code.cpp
main()
1 #include <iostream>
2 using namespace std;
3
4 int factorial(int n) {
5     int fact = 1;
6
7     for(int i=1; i<=n; i++) {
8         fact *= i;
9     }
10    return fact;
11 }
12
13 int main() {
14     int n = 8, r = 2;
15
16     int fact_n = factorial(n);
17
18     return 0;
19 }
20
apnacollege@Shradha DSAseries %
```



```
code.cpp
main()
6
7     for(int i=1; i<=n; i++) {
8         fact *= i;
9     }
10    return fact;
11 }
12
13 int nCr(int n, int r) {
14     int fact_n = factorial(n);
15     int fact_r = factorial(r);
16     int fact_nmr = factorial(n-r);
17
18     return fact_n / (fact_r * fact_nmr);
19 }
20
21 int main() {
22     int n = 6, r = 3;
23     cout << nCr(n, r) << endl;
24
25
26     return 0;
27 }
28
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
28
apnacollege@Shradha DSAseries % g++ code.cpp && ./a.out
20
apnacollege@Shradha DSAseries %
```



functions — return
— param/arguments
fact, sum, bin coeff, digits sum
fun memory
call by value

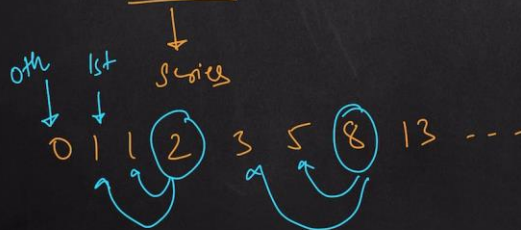


Homework

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.



switch () {

}



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;

//}
```