
1- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    bool isPositiveResult = isPositive(10);
    std::cout << "Is Positive? " << std::boolalpha << isPositiveResult << "\n";

    bool isNegativeResult = isNegative(-7);
    std::cout << "Is Negative? " << std::boolalpha << isNegativeResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

bool isPositive(int number) {
    return (number > 0);
}

bool isNegative(int number) {
    return (number < 0);
}
```

Solution

```
Hello, welcome to the program!
Is Positive? true
Is Negative? true
```

2- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    double averageOfArrayResult = calculateAverageOfArray({10, 20, 30, 40, 50});
    std::cout << "Average of Array: " << averageOfArrayResult << "\n";

    bool isLeapYearResult = isLeapYear(2024);
    std::cout << "Is Leap Year? " << std::boolalpha << isLeapYearResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>
#include <vector>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

double calculateAverageOfArray(const std::vector<int>& numbers) {
    int sum = 0;
    for (int num : numbers) {
        sum += num;
    }
    return static_cast<double>(sum) / numbers.size();
}

bool isLeapYear(int year) {
    return ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0));
}
```

Solution

```
Hello, welcome to the program!
Average of Array: 30
Is Leap Year? true
```

3- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    int maxOfThreeResult = findMaxOfThree(8, 12, 6);
    std::cout << "Max of Three: " << maxOfThreeResult << "\n";

    int minOfThreeResult = findMinOfThree(25, 17, 30);
    std::cout << "Min of Three: " << minOfThreeResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

int findMaxOfThree(int a, int b, int c) {
    return (a > b) ? ((a > c) ? a : c) : ((b > c) ? b : c);
}

int findMinOfThree(int a, int b, int c) {
    return (a < b) ? ((a < c) ? a : c) : ((b < c) ? b : c);
}
```

Solution

```
Hello, welcome to the program!
Max of Three: 12
Min of Three: 17
```

4- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    int cubeResult = calculateCube(4);
    std::cout << "Cube: " << cubeResult << "\n";

    bool isZeroResult = isZero(0);
    std::cout << "Is Zero? " << std::boolalpha << isZeroResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

int calculateCube(int x) {
    return x * x * x;
}

bool isZero(int number) {
    return (number == 0);
}
```

Solution

```
Hello, welcome to the program!
Cube: 64
Is Zero? true
```

5- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    int doubleResult = calculateDouble(9);
    std::cout << "Double: " << doubleResult << "\n";

    bool isOddResult = isOdd(11);
    std::cout << "Is Odd? " << std::boolalpha << isOddResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

int calculateDouble(int x) {
    return x * 2;
}

bool isOdd(int number) {
    return (number % 2 != 0);
}
```

Solution

```
Hello, welcome to the program!
Double: 18
Is Odd? true
```

6- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    int negateResult = negateNumber(7);
    std::cout << "Negate: " << negateResult << "\n";

    bool isPositiveResult = isPositive(10);
    std::cout << "Is Positive? " << std::boolalpha << isPositiveResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

int negateNumber(int x) {
    return -x;
}

bool isPositive(int number) {
    return (number > 0);
}
```

Solution

```
Hello, welcome to the program!
Negate: -7
Is Positive? true
```

7- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    int absoluteResult = calculateAbsolute(5);
    std::cout << "Absolute: " << absoluteResult << "\n";

    bool isNegativeResult = isNegative(-3);
    std::cout << "Is Negative? " << std::boolalpha << isNegativeResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

int calculateAbsolute(int x) {
    return (x < 0) ? -x : x;
}

bool isNegative(int number) {
    return (number < 0);
}
```

Solution

```
Hello, welcome to the program!
Absolute: 5
Is Negative? true
```

8- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    int negateResult = negateNumber(7);
    std::cout << "Negate: " << negateResult << "\n";

    bool isEvenResult = isEven(14);
    std::cout << "Is Even? " << std::boolalpha << isEvenResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

int negateNumber(int x) {
    return -x;
}

bool isEven(int number) {
    return (number % 2 == 0);
}
```

Solution

```
Hello, welcome to the program!
Negate: -7
Is Even? true
```

9- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    int squareResult = calculateSquare(7);
    std::cout << "Square: " << squareResult << "\n";

    bool isZeroResult = isZero(0);
    std::cout << "Is Zero? " << std::boolalpha << isZeroResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

int calculateSquare(int x) {
    return x * x;
}

bool isZero(int number) {
    return (number == 0);
}
```

Solution

```
Hello, welcome to the program!
Square: 49
Is Zero? true
```

10- Trace the following program and predict the output.

```cpp
// www.gammal.tech

// main.cpp
#include <iostream>
#include "functions.h"

int main() {
    printMessage();

    bool isEvenResult = isEven(14);
    std::cout << "Is Even? " << std::boolalpha << isEvenResult << "\n";

    bool isPrimeResult = isPrime(13);
    std::cout << "Is Prime? " << std::boolalpha << isPrimeResult << "\n";

    return 0;
}
```

```cpp
// www.gammal.tech

// functions.cpp
#include <iostream>

void printMessage() {
    std::cout << "Hello, welcome to the program!\n";
}

bool isEven(int number) {
    return (number % 2 == 0);
}

bool isPrime(int number) {
    if (number < 2) {
        return false;
    }
    for (int i = 2; i * i <= number; ++i) {
        if (number % i == 0) {
            return false;
        }
    }
    return true;
}
```

Solution

```
Hello, welcome to the program!
Is Even? true
Is Prime? true
```