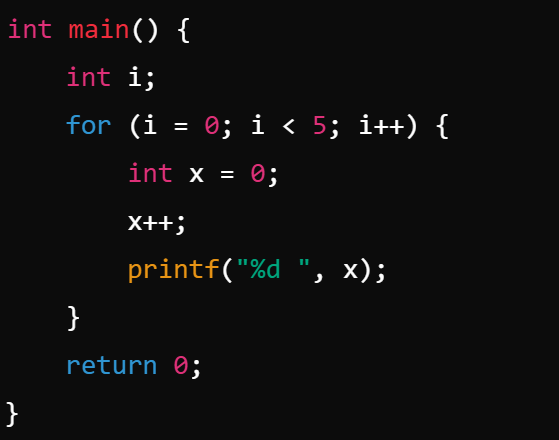
### **THEORY (10 marks)**

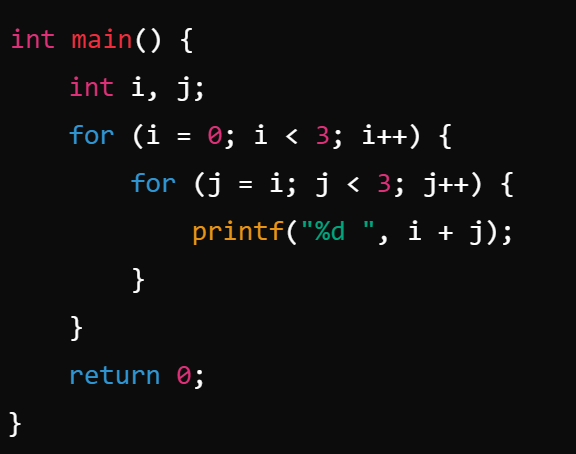
1. **What will be the output of the following code? (2 marks)**



A. 1 1 1 1 1 B. 0 1 2 3 4

C. 1 2 3 4 5 D. 1 2 3 4 0

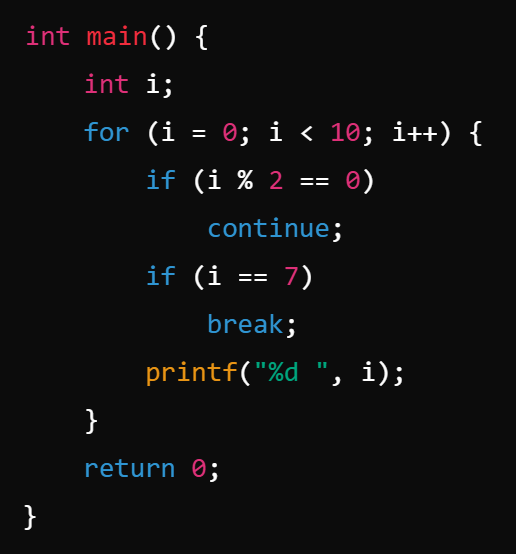
1. **What will be the output of the following code? (2 marks)**



A. 0 1 2 1 2 2 B. 0 1 2 2 3 4

C. 0 1 2 1 2 D. 0 1 2 1 2 2 3

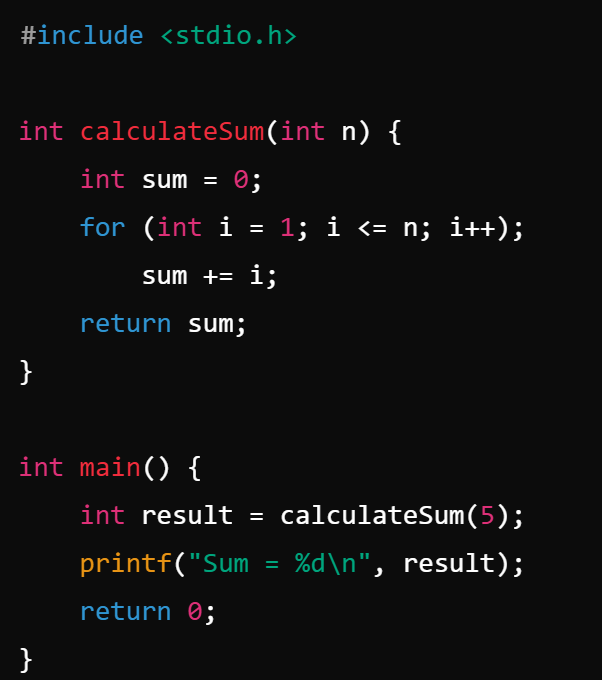
1. **What is the output of the following program? (2 marks)**



A. 1 3 5 B. 1 3 5 7

C. 1 2 3 4 5 6 D. 1 3 5 7 9

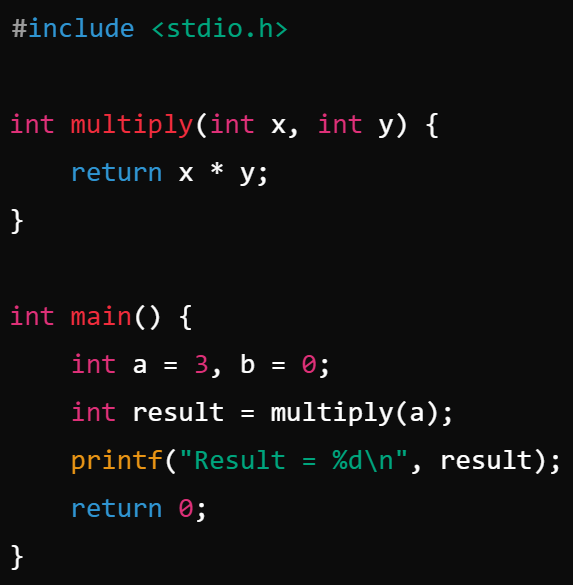
1. **Consider the following C code snippet: (2 marks)**



A. Sum = 15 B. Sum = 0

C. Sum = 6 D. The program will not compile

1. **Consider the following C code snippet: (2 marks)**



A. Result = 3 B. Result = 0

C. Result = Garbage value D. Other result

### 

### **Exercise: Prime Number Counter (10 marks)**

Write a C program that includes two functions:

1. isPrime(int n): This function takes an integer n as input and returns 1 if n is a prime number and 0 otherwise. A prime number is a number greater than 1 that has no divisors other than 1 and itself. (4 marks)
2. countPrimes(int start, int end): This function takes two integers start and end as inputs and returns the number of prime numbers between start and end (inclusive). (3 marks)

The **main** function should ask the user to input two integers and then call the countPrimes() function to count how many prime numbers are in that range, finally printing the result. (3 marks)

### **Example:**

For input:

Enter the range (start and end): 10 30

The output should be:

There are 6 prime numbers between 10 and 30.

Prime numbers between 10 and 30 are: 11, 13, 17, 19, 23, 29

