JAVA TEST - 2

Part 1 : MCQ Questions

1. Swapping Two Numbers

```
public class SwapWith3rdVar {
    public static void main(String[] args) {
        int a = 5;
        int b = 3;
        System.out.println("Before swapping a and b are: " + a + " , " + b);

        // Missing code here

        System.out.print("After swapping a and b are: " + a + " , " + b);
    }
}
```

Which of the following options correctly swaps the values of a and b without using a third variable?

- **A.** `a = b; b = a;`
- **B.** `a = a + b; b = a - b; a = a - b;`
- **C.** `int temp = a; a = b; b = temp;`
- **D.** `a = a * b; b = a / b; a = a / b;`

2. Reversing a Number

```
public class Reverse_Number {
    public static void main(String[] args) {
        int n = 6523;
        int rev = 0;
        int rem = 0;
        while (n > 0) {
            rem = n % 10;
            // Missing line here
            n /= 10;
        }
        System.out.println(rev);
    }
}
```

Which line of code should be added to correctly reverse the number?

- A. `rev = rev + rem;`
- B. `rev = rev * 10 + rem;`
- C. `rev = rem * 10 + rev;`
- D. `rev = rem + rev * 10;`

3. Prime Numbers from 1 to 25

```
public class PrimeNumber {
    public static void main(String[] args) {
        for (int n = 1; n <= 25; n++) {
            Boolean prime = true;
            for (int i = 2; i <= n / 2; i++) {
                if (n % i == 0) {
                    prime = false;
                    break;
                }
            }
            // Missing condition here
            {
                System.out.println(n);
            }
        }
    }
}
```

Which condition should be placed to print only the prime numbers?

- A. `if (prime == false)`
- B. `if (prime)`

- C. `if (n > 1 && prime)`
- D. `if (n == 2 || prime)`

4. Simple Interest Calculation

```java
public class SimpleInterest {
    public static void main(String[] args) {
        float p = 5600;
        float r = 0.5f;
        float t = 2;
        float num = p * r * t;
        // Missing line here
    }
}
```

Which line of code should be added to correctly calculate and display the Simple Interest?

- A. `System.out.println("Si is :" + num);`
- B. `float si = num; System.out.println("Si is :" + si);`
- C. `float si = num / 100; System.out.println("Si is :" + si);`
- D. `float si = num * 100; System.out.println("Si is :" + si);`

5. Fibonacci Series

```java
public class Fibonacci_Series {
    public static void main(String[] args) {
        int n = 7;
        int first = 0;
        int second = 1;
        System.out.print(first + " " + second + " ");
        for (int i = 2; i < n; i++) {
            // Missing line here
            System.out.print(third + " ");
            first = second;
            second = third;
        }
    }
}
```

Which line of code should be added to generate the Fibonacci series correctly?

- A. `int third = first - second;`
- B. `int third = first * second;`
- C. `int third = first + second;`
- D. `int third = second - first;`

6. Nested Loop with Break

```
public class BreakExample {
    public static void main(String[] args) {
        for (int i = 1; i <= 3; i++) {
            for (int j = 1; j <= 3; j++) {
                if (i == j) break;
                System.out.print(i * j + " ");
            }
        }
    }
}
```

What is the output of the above code?

- A. 1 2 2 4 3 6
- B. 2 4 6
- C. 1 2 4 6
- D. 2 6

7. Function with Return Statement

```
public class ReturnExample {

    static int max(int x, int y) {

        if (x > y) return x;

        else return y;

    }

    public static void main(String[] args) {

        int a = 3, b = 5;

        System.out.println(max(a, b));

    }

}
```

What is the output of the above code?

- **A. 3**
- **B. 5**
- **C. 8**
- **D. Compile-time Error**

8. Which of the following is true about functions in Java?

- A. Functions can return multiple values.
- B. Functions can be called within loops.
- C. Functions cannot call other functions.
- D. Functions do not support recursion.

9. What is recursion in programming?

Recursion is a technique where:

- A. A function calls another function.
- B. A function calls itself.
- C. A loop is repeated multiple times.
- D. A function runs indefinitely.

10. What is the main advantage of using recursion?

- A. Faster execution time.
- B. Simplifies complex problems.
- C. Uses less memory.
- D. No need for base condition.

Part 2 : Coding Questions

1. **Calculate Power**
   Write a function to calculate the power of a number using nested loops.
   Example:
   Input: `base = 2, exponent = 3`
   Output: `8`

---

2. **Menu-Driven Calculator**
   Create a menu-driven calculator using switch case for the following operations:

- Addition
- Subtraction
- Multiplication
- Division

The user should input two numbers and an operator. The program should display the result of the chosen operation.

3. **Check Strong Number**
   Write a function to check if a number is a Strong number.
   (A Strong number is a number for which the sum of the factorials of its digits equals the number itself.)
   Example:
   Input: `145`
   Output: `Strong Number` (since 1!+4!+5!=1451! + 4! + 5! = 1451!+4!+5!=145)

4. **Sum of Alternate Digits**
   Calculate the sum of alternate digits of a number using nested loops.
   Example:
   Input: `12345`
   Output: `9` (i.e., 1+3+51 + 3 + 51+3+5)

5. **Hourglass Pattern**
   Write a program to print an hourglass pattern for a given number of rows.
   Example for `n = 4`:

```
1 2 3 4 5 6 7
```

```
 2 3 4 5 6 7
```

```
  3 4 5 6 7
   4 5 6 7
    5 6 7
     6 7
      7
     6 7
    5 6 7
   4 5 6 7
  3 4 5 6 7
 2 3 4 5 6 7
1 2 3 4 5 6
```

6. Find Second Maximum in an Array

Write a program to find the second maximum element in an array.

7. Binary to Decimal Conversion using Recursion

Write a recursive function to convert a binary number to its decimal equivalent.

.