**DE NOBILI SCHOOL KORADIH**

**SUMMER VACATION HOMEWORK(2025-26)**

**Class – 11 subject :- Computer Science**

**Question 1:**

1. **Convert the following binary numbers to decimal:**  
   a) 1010  b) 111010  c) 101011111
2. **Convert the following decimal numbers to binary:**  
   a) 25  b) 58  c) 127
3. **Convert the following decimal numbers to octal:**  
   a) 64  b) 100  c) 255
4. **Convert the following decimal numbers to hexadecimal:**  
   a) 45  b) 90  c) 255
5. **Convert the following hexadecimal numbers to binary:**  
   a) A2  b) 1F  c) 7B
6. **Convert the following binary numbers to hexadecimal:**  
   a) 1101  b) 10110100  c) 11111111
7. **Convert the following octal numbers to decimal:**  
   a) 17  b) 75  c) 377
8. **Convert the following hexadecimal numbers to decimal:**  
   a) 2A  b) 3E  c) FF
9. **Convert the following binary numbers to octal:**  
   a) 101000  b) 110111  c) 1001001
10. **Convert the following octal numbers to binary:**  
    a) 12  b) 45  c) 71

Question 2.

1.  **Perform the following binary additions:**  
   a) 1011 + 1100 b) 1001 + 0110 c) 1110 + 0011
2.  **Perform the following binary subtractions using 1's complement method:**  
   a) 1101 – 1000 b) 1010 – 0111 c) 1000 – 1011
3.  **Perform the following binary subtractions using 2's complement method:**  
   a) 1101 – 1010 b) 1111 – 0110 c) 1011 – 1100
4.  **Perform the following binary multiplications:**  
   a) 101 × 11 b) 110 × 10 c) 111 × 100
5.  **Perform the following binary divisions:**  
   a) 1100 ÷ 10 b) 10010 ÷ 101 c) 1111 ÷ 11
6.  **Perform the following octal additions:**  
   a) 17 + 21 b) 34 + 45 c) 52 + 16
7.  **Perform the following octal subtractions using borrowing:**  
   a) 65 – 27 b) 51 – 35 c) 76 – 45
8.  **Perform the following hexadecimal additions:**  
   a) A5 + 3C b) 7F + 1A c) 9D + 62
9.  **Perform the following hexadecimal subtractions:**  
   a) 8C – 2A b) 7D – 3F c) FF – B4
10.  **Convert and multiply the following hexadecimal numbers (show the intermediate steps in binary):**  
    a) A × 3 b) 7 × 5 c) C × 4

**Question 3.**

1. **Write a class program. Class name: Number**  
   **Data member:**
   * int num

**Member functions:**

* + number(): to initialize num with 0 // default constructor
  + void input(): to accept a number
  + void checkArmstrong(): to check whether the number is an Armstrong number  
    **Task:** Display an appropriate message whether the number is Armstrong or not.

1. **Write a class program. Class name: Palindrome**  
   **Data member:**
   * int number

**Member functions:**

* + Palindrome(): to initialize number with 0 // default constructor
  + void input(): to accept a number
  + void isPalindrome(): to check whether the number is a palindrome  
    **Task:** Display whether the number is a palindrome or not.

1. **Write a class program. Class name: PrimeCheck**  
   **Data member:**
   * int n

**Member functions:**

* + PrimeCheck(): to initialize n with 0 // default constructor
  + void input(): to accept a number
  + void checkPrime(): to determine if the number is prime  
    **Task:** Display whether the number is prime or not.

1. **Write a class program. Class name: SeriesSum**  
   **Data member:**
   * int limit

**Member functions:**

* + SeriesSum(): to initialize limit with 0 // default constructor
  + void input(): to accept the range limit
  + void calculateSum(): to calculate the sum of the series 1 + 2 + 3 + ... + n  
    **Task:** Display the total sum.

1. **Write a class program. Class name: FactorialCalc**  
   **Data member:**
   * int value

**Member functions:**

* + FactorialCalc(): to initialize value with 0 // default constructor
  + void input(): to accept a number
  + void findFactorial(): to find the factorial of the number  
    **Task:** Display the factorial result.

1. **Write a class program. Class name: EvenOddCounter**  
   **Data members:**
   * int start, end

**Member functions:**

* + EvenOddCounter(): to initialize start and end with 0 // default constructor
  + void input(): to accept a range
  + void countEvenOdd(): to count and display even and odd numbers in the range  
    **Task:** Display the count of even and odd numbers.

1. **Write a class program. Class name: FibonacciSeries**  
   **Data member:**
   * int terms

**Member functions:**

* + FibonacciSeries(): to initialize terms with a default value
  + void input(): to accept number of terms
  + void printFibonacci(): to generate and print the Fibonacci series  
    **Task:** Display the generated Fibonacci series.

1. **Write a class program. Class name: DigitCounter**  
   **Data member:**
   * int num

**Member functions:**

* + DigitCounter(): to initialize num with 0 // default constructor
  + void input(): to accept a number
  + void countDigits(): to count and display the number of digits  
    **Task:** Display how many digits the given number has.

1. **Write a class program. Class name: TableGenerator**  
   **Data member:**
   * int num

**Member functions:**

* + TableGenerator(): to initialize num with 0 // default constructor
  + void input(): to accept a number
  + void printTable(): to print the multiplication table up to 10  
    **Task:** Display the table of the given number.

1. **Write a class program. Class name: ReverseNumber**  
   **Data member:**

* int n

**Member functions:**

* ReverseNumber(): to initialize n with 0 // default constructor
* void input(): to accept a number
* void reverse(): to reverse the digits of the number. 786 --> 687   
  **Task:** Display the reversed number.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*