INDIAN INSTITUTE OF TECHNOLOGY TIRUPATI
DEPARTMENT OF MATHEMATICS AND STATISTICS

Class Test - 6 **MA517M-Basic Programming Laboratory** 22 September 2025

Name **Roll No.: MA25M**

Problem (with Functions, Arrays, and Pointers)

In number theory, a **heptaphobic number** is defined as follows:

> A **positive integer** is called *heptaphobic* if:
>
> 1. It is **not divisible by 7**, and
> 2. **No number divisible by 7** can be produced by **swapping any two digits** of the number (without creating leading zeros).

Examples

- **17** is heptaphobic: Not divisible by 7. Swapping digits gives 71, which is not divisible by 7. ✓
- **103** is heptaphobic: Not divisible by 7. Swapping digits (130, 301, etc.) does not give a multiple of 7. ✓
- **14** is **not** heptaphobic: It is divisible by 7. ×
- **231** is **not** heptaphobic: Not divisible by 7, but swapping gives 213, which **is** divisible by 7. ×

Task

Write a **modular C++ program** that makes use of **functions, arrays, and pointers** to:

1. Read an integer `N` from the user.
2. Count and print the total number of **heptaphobic numbers strictly less than** `N`.
3. Your program must:
   - Store the digits of a number in an **array**.
   - Use **functions** for tasks such as:
     (a) Checking divisibility by 7.
     (b) Generating all valid digit swaps (excluding numbers with leading zeros).
     (c) Checking if a number is heptaphobic.
   - Pass arrays to functions using **pointers**.
   - Ensure efficiency for reasonably large values of `N` (e.g., up to 10,000).

Expected Output

```
Total heptaphobic numbers less than N: <count>
```

1. Write a C++ program to add two vectors using pointers
2. Write a C++ function to find the GCD and LCM of two numbers using functions (you can also use recursion). Use it to find gcd of 4 numbers.

INDIAN INSTITUTE OF TECHNOLOGY TIRUPATI
DEPARTMENT OF MATHEMATICS AND STATISTICS

Class Test - 6 **MA517M-Basic Programming Laboratory** 22 September 2025
Name **Roll No.: MA25M**

Truncatable Primes (Using Functions, Arrays, and Pointers)

A prime number is called **truncatable** if it remains prime when digits are successively removed from either the left or the right.

Definition

A number is a **truncatable prime** if:

1. The number itself is prime.
2. All numbers obtained by successively removing digits from **left to right** are prime.
3. All numbers obtained by successively removing digits from **right to left** are prime.

Note: The single-digit primes (2, 3, 5, and 7) are **not considered** truncatable primes.

Example

Consider the number 3797:

- **Left to right truncation:** 3797, 797, 97, 7 — all are prime. ✓
- **Right to left truncation:** 3797, 379, 37, 3 — all are prime. ✓

Therefore, 3797 is a **truncatable prime**.

Task

Write a modular **C++ program** that makes use of **functions, arrays, and pointers** to:

1. Implement a function to check whether a number is prime.
2. Store the digits of a number in an **array**, and use **pointers** to pass the array to helper functions.
3. Implement separate functions to:
   (a) Truncate digits from left to right using the digit array.
   (b) Truncate digits from right to left using the digit array.
   (c) Check if a number is truncatable prime (using the above functions).
4. Identify all numbers that are truncatable primes (there are exactly **eleven** such primes).
5. Compute and print the **sum** of these eleven truncatable primes.

Output Format

```
11 Truncatable Primes: 23, 37, 53, 73, 313, 317, 373, 797, 3137, 3797, 739397
Sum of the eleven truncatable primes: 748317
```

1. Write a C++ program to add two vectors using pointers
2. Write a C++ function to find the GCD and LCM of two numbers using functions (you can also use recursion). Use it to find gcd of 4 numbers.

INDIAN INSTITUTE OF TECHNOLOGY TIRUPATI
DEPARTMENT OF MATHEMATICS AND STATISTICS

Class Test - 6 **MA517M-Basic Programming Laboratory** 22 September 2025
Name **Roll No.: MA25M**

Circular Primes (Using Functions, Arrays, and Pointers)

A prime number is called a **circular prime** if all rotations of its digits are also prime numbers.

Definition

A number is a **circular prime** if:

1. The number itself is prime.
2. All cyclic rotations of its digits also result in prime numbers.

For example, the number `197` is a circular prime because:

- Rotations: `197`, `971`, `719`
- All three numbers are prime.

Task

Write a modular **C++ program** that makes use of **functions, arrays, and pointers** to:

1. Implement a function to check whether a number is prime.
2. Extract and store the digits of a number in an **array**, and pass this array to functions using **pointers**.
3. Implement a function to generate all **cyclic rotations** of a number using the digit array and pointers.
4. Implement a function that checks whether a number is a **circular prime** (using prime-checking and rotations).
5. Find all circular prime numbers less than `1,000`.
6. Count and display the total number of circular primes found.

Constraints

- Use efficient prime checking, suitable for values up to one million.
- Use arrays and pointer-based digit manipulation for generating rotations.

Expected Output

```
Total number of circular primes below 1,000: <count>
```

Example

Below 100, the circular primes are:

$$2,\ 3,\ 5,\ 7,\ 11,\ 13,\ 17,\ 31,\ 37,\ 71,\ 73,\ 79,\ 97$$

Total: 13 circular primes under 100.

1. Write a C++ program to add two vectors using pointers
2. Write a C++ function to find the GCD and LCM of two numbers using functions (you can also use recursion). Use it to find gcd of 4 numbers.