# Programming Fundamentals – Lab 10

**Topics to be covered**

- Pointer arithmetic
- Arrays as pointers
- File handling with pointers
- Dynamic single variable allocation
- Functions returning pointers

## Q1: Sum of Alternate Elements

Write a C++ program that:
- Reads an integer `n` (maximum 100).
- Reads `n` integers into  array.
- Calculates and prints the sum of every alternate element starting from the first using pointer arithmetic only.

```
Example Input:
5
5 10 15 20 25
```

Expected Output:
Sum of alternate elements = 45

## Q2: Max-Min Difference from File

Write a C++ program that:
- Reads integers from a file named `data.txt` into  array .
- Finds and prints the difference between the maximum and minimum values using pointer arithmetic only.
- Handles file open failure gracefully.

```
Example File Content (data.txt):
12 45 7 89 34
```

Expected Output:
Max difference = 82

## Q3: Filter Even Numbers

Write a C++ program that:
- Reads an integer `n` (maximum 100), then reads `n` integers into a normal array.
- Copies even numbers into another normal array.

- Prints the even numbers and their count using pointer arithmetic only.
- Does not modify the original array.

```
Example Input:
5
3 6 9 12 15
```

Expected Output:
Even numbers: 6 12
Count: 2

### Q4: Reverse Array from File

Write a C++ program that:
- Reads integers from a file named `nums.txt` into a normal array (maximum 100 values).
- Reverses the array using pointer arithmetic only.
- Writes the reversed array to a file named `reversed.txt`.
- Writes the count of numbers on a new line at the end of the file.

```
Example Input File (nums.txt):
2 4 6 8
```

Expected Output File (reversed.txt):
8 6 4 2
Count: 4

### Q5: Scale Elements Based on First

Write a C++ program that:
- Reads an integer `n` (maximum 100), then reads `n` integers into a normal array.
- Modifies the array such that each element is scaled by dividing it by the first element and multiplying by 10.
- Uses pointer arithmetic only to access and modify elements.
- Prints the updated array.

```
Example Input:
3
5 10 15
```

Expected Output:
Updated array: 10 20 30

### Q6: Dynamic Allocation

Write a C++ program that:
- Dynamically allocates memory for a single integer.
- Prompts the user to enter a number.

- Multiplies the value by 5 and displays the result.
- Frees the allocated memory after use.

```
Example Input:
Enter a number: 7
```

Expected Output:
Result = 35

### Q7: Function Returning a Pointer

Write a C++ program that:
- Defines a function `findMaxPointer(int* arr, int size)` that returns a pointer to the maximum element in the array.
- In `main()`, read `n` (maximum 100) and `n` integers into a normal array.
- Call the function and print the value of the maximum element using the returned pointer.
- Use pointer arithmetic in the function to find the maximum.

```
Example Input:
6
11 28 47 36 9 24
```

Expected Output:
Maximum element is: 47