LAB 3B

COMPUTER PROGRAMMING

Topics for Today

1. Break/Continue
2. Type Casting
3. Static vs Global Variables
4. Arrays

Break and Continue Statements

- **Break and Continue in Loops:**
- **Break:** Terminates the loop prematurely.
  - Example: Exiting a loop when a certain condition is met.
- **Continue:** Skips the current iteration and moves to the next one.
  - Example: Skipping even numbers in a loop that processes only odd numbers.

```c
for (int i = 0; i < 10; i++) {
    if (i == 5) break;  // Exit loop when i equals 5
    if (i % 2 == 0) continue;  // Skip even numbers
    printf("%d ", i);  // Only prints odd numbers
}
```

Type Casting

- **Type Casting:**
- **Implicit Casting:** Automatic conversion of data types (e.g., `int` to `float`).
- **Explicit Casting:** Manual conversion of data types using type casting operators.
  - Example: Converting `float` to `int` to truncate decimal values.

```
float num = 9.7;
int truncated = (int)num;  // Explicit casting from float to int
```

- **Use Case:** Handling mixed data types in arrays and arithmetic operations.

Static vs Global Variables

Static Variables:

- **Static inside functions:** Retains value between function calls.
- **Static outside functions:** Limits the scope to the file, similar to global but restricted to that file.

```
void staticExample() {
    static int count = 0;  // Retains value between function calls
    count++;
    printf("%d\n", count);
}
```

Static vs Global Variables

Global Variables:

- Declared outside all functions and accessible to all functions in the program.

```
int globalVar = 10;
```

- **Use Case:** Control variable scope and lifetime effectively.

Arrays

Array Basics:

- **Definition:** An array is a collection of elements of the same data type stored at contiguous memory locations.

```
int arr[5];  // Declaring an integer array of size 5
```

- **Accessing Elements:** Array elements are accessed using indices, starting from 0.

```
arr[0] = 10;  // Assigns value 10 to the first element
```

Row-Major and Column-Major Forms

Row-Major vs Column-Major:

- **Row-Major Order:** Elements of a 2D array are stored row by row in memory.
  - Example: `arr[0][0]` is followed by `arr[0][1]`.
- **Column-Major Order:** Elements are stored column by column in memory.
  - Example: `arr[0][0]` is followed by `arr[1][0]`.
- **Row-Major:**

- **Column-Major:**

Problem on Arrays: Reversing the Array

Reversing an Array:

- **Concept:** Swap elements from the beginning and end of the array, moving towards the center.

```
void reverseArray(int arr[], int N) {
    for (int i = 0; i < N/2; i++) {
        int temp = arr[i];
        arr[i] = arr[N-i-1];
        arr[N-i-1] = temp;
    }
}
```

Questions?

LET'S START CODING !