# 3 Lecture - CS410

# **Important Subjective**

#### 1. Question: What is an array in C?

**Answer**: An array in C is a collection of elements of the same data type, stored in contiguous memory locations and accessed using an index.

#### 2. Question: How do you declare an array in C?

**Answer**: You declare an array in C using the syntax: "data\_type array\_name[size];"

#### 3. Question: Explain the concept of pointers in C.

**Answer**: Pointers are variables that store memory addresses. They are used to manipulate memory directly and are useful for dynamic memory allocation.

#### 4. Question: How do you declare a pointer in C?

**Answer**: To declare a pointer in C, you use the syntax: "data\_type \*pointer\_name;"

#### 5. Question: What does the "sizeof" operator do with arrays and pointers?

**Answer**: The "sizeof" operator returns the size of an array in bytes and the size of a pointer in the specific platform's memory address length.

#### 6. Question: How do you access elements of an array using pointers in C?

**Answer**: You can access array elements using pointers by dereferencing the pointer and using the array index notation, like "\*(ptr + index)".

#### 7. Question: What happens when you pass an array to a function in C?

**Answer**: When an array is passed to a function, it decays into a pointer, and the function receives a pointer to the first element.

#### 8. Question: How do you dynamically allocate memory for an array using pointers in C?

**Answer**: You can use functions like "malloc" or "calloc" to dynamically allocate memory for an array and assign the memory address to a pointer.

## 9. Question: How do you deallocate memory for a dynamically allocated array in C?

**Answer**: You use the "free" function to deallocate memory for a dynamically allocated array.

## 10. Question: Explain the concept of pointer arithmetic in C.

**Answer**: Pointer arithmetic involves performing arithmetic operations (like addition or subtraction) on pointers. This allows moving to different memory locations based on the pointer's data type, size, and the operation performed.