18-12-2025

A. Question And Answers

1. **What is C language and why is it called a middle-level language?**

   C is a procedural programming language developed by Dennis Ritchie.
   It is called a **middle-level language** because it supports both **low-level features** (pointers, memory access) and **high-level features** (functions, loops, structures).

2. **What is a pointer? Why is it used?**

   A pointer is a variable that **stores the address of another variable**.
   It is used for **dynamic memory allocation, passing values by reference, and efficient memory management**.

3. **What is the difference between malloc() and calloc()?**

Malloc()	**Calloc()**
Allocates single block	Allocates multiple blocks
Does not initialize memory	Initializes memory to 0
Faster	Slightly slower

4. **What is the difference between structure and union?**

Structure	**Union**
Each member has its own memory	All members share same memory
More memory usage	Less memory usage
All members accessible	One member at a time

5. **What is segmentation fault?**

   A segmentation fault occurs when a program tries to access invalid memory, such as:
   - Using uninitialized pointers
   - Accessing array out of bounds

6. **What is the difference between call by value and call by reference?**

   - **Call by value:** Copy of variable is passed → original value not changed
   - **Call by reference:** Address is passed using pointers → original value changes

7. **What is an array? How is it stored in memory?**

An array is a collection of **same data type elements** stored in **contiguous memory locations**.

8. **What is the difference between array and pointer?**

Array	Pointer
Fixed size	Can be dynamic
Stores values	Stores address
Cannot be reassigned	Can be reassigned

9. **What are storage classes in C?**

Storage classes define scope, lifetime, and visibility of variables.

Types:

- auto
- static
- extern
- register

10. **What is a dangling pointer?**

A dangling pointer points to a **memory location that has been freed or deleted**.
It can cause **unexpected behavior or crashes**.

B. Explain about storage classes.

Storage Classes in C Programming

Definition

Storage classes in C define **four important properties of a variable**:

- **Scope** – where the variable can be accessed
- **Lifetime** – how long the variable exists in memory
- **Visibility** – which parts of the program can see it
- **Default value** – initial value if not assigned

Types of Storage Classes in C

C supports **four storage classes**:

1. auto
2. register
3. static
4. extern

1. auto Storage Class

- ➢ Default storage class for **local variables**
- ➢ Stored in **stack memory**
- ➢ Scope: Inside the block only
- ➢ Default value: **Garbage**

Example:

```
void main() {
   auto int x = 10;
}
```

- ◆ *Note:* Writing auto is optional

2. register Storage Class

- ➢ Suggests storing variable in **CPU register**
- ➢ Faster access than memory
- ➢ Address (&) **cannot be used**

Example:

```
register int count;
```

- ◆ Used in **loops and counters**

3. static Storage Class

- ➢ Retains value **even after function execution**
- ➢ Initialized only **once**
- ➢ Stored in **data segment**
- ➢ Default value: **0**

Example:

```
 void fun() {

   static int x = 0;

   x++;

   printf("%d ", x);

}
```

Output: 1 2 3 (on repeated calls)

- ◆ Used when data persistence is needed

4. **extern Storage Class**

   - ➢ Used to access global variables defined in another file
   - ➢ Does not allocate memory (only declaration)

Example:

```
extern int a;
```

- ◆ Used in **multi-file programs**

Comparison Table

Storage Class	**Scope**	**Lifetime**	**Default Value**	**Memory**
auto	Block	Block	Garbage	Stack
register	Block	Block	Garbage	CPU Register
static	Block/file	Entire program	0	Data Segment
extern	Global	Entire program	0	Data Segment