Assignment 31: What are different storage classes in C programming ?

In C programming, there are several storage classes, each defining the scope (visibility) and lifetime of variables declared within them. The main storage classes in C are:

- auto: This is the default storage class for all local variables. Variables declared with the auto storage class are automatically created and initialized every time the block they are declared in is entered, and they are destroyed when the block is exited.

- register: This storage class is used to define local variables that should be stored in CPU registers, if possible, for faster access. However, the register keyword is a hint to the compiler, and it's up to the compiler to decide whether to honor it or not.

- static: Variables declared with the static storage class retain their values between function calls. They are initialized only once and keep their values throughout the program's execution. Additionally, static variables have file scope when declared outside of any function, meaning they are accessible only within the file they are declared in.

- extern: The extern storage class is used to declare variables that are defined in another file of the program or in a library. It essentially tells the compiler that the variable is declared elsewhere, and the actual definition will be provided at linking time.

- typedef: While not a storage class in the traditional sense, typedef is used to create aliases for data types, making code more readable and portable. It does not affect storage duration or scope.

- Thread Local Storage (TLS): Introduced in C11, TLS allows variables to have separate instances for each thread, ensuring thread safety.

- These storage classes provide flexibility in managing the lifetime and scope of variables in C programs.