Assignment 65: What are Unformatted Input/Output Functions in C?

Unformatted Input/Output (I/O) functions in C deal with raw input/output without any specific formatting. These functions typically read or write individual characters or strings without considering formatting options. Here are some of the common unformatted input/output functions in C:

❖ Character Input/Output Functions:

- getchar(): Reads a single character from the standard input (usually the keyboard).
- putchar(): Writes a single character to the standard output (usually the console).
- getch(): Reads a single character from the standard input without echoing it to the screen.
- getche(): Reads a single character from the standard input and echoes it to the screen immediately.
- These functions are used for basic character-based input/output operations without any specific formatting.

❖ String Input/Output Functions:

- gets(): Reads a string from the standard input (deprecated due to security concerns).
- puts(): Writes a string to the standard output followed by a newline character.
- fgets(): Reads a string from the standard input with specified buffer size (safer alternative to gets()).
- These functions are used for input/output operations involving strings (sequences of characters) without any specific formatting.

Unformatted input/output functions are straightforward and simple to use but may lack the ability to control the formatting of input/output data. They are commonly used for basic input/output tasks where formatting is not a concern, such as reading individual characters or processing raw string data. However, it's important to use them carefully to avoid buffer overflow and security vulnerabilities, especially in the case of string input functions like gets(), which is deprecated in newer versions of the C standard.