1. **Define the following terms.**
2. Compiler - Refers to a software program that converts the source code written in high level language into machine code that the computer can execute.
3. Object code - Refers to machine code generated by a compiler or assembler.
4. Source code - Refers to code written in high level language.
5. Linkers - Refers to a program that takes on or more object files generated by a compiler and combines them into a single executable program.
6. **Using an example, write a program to add two numbers, explain the compilation process of c program.**

#include <stdio.h>

int main() {

// Declare variables

int num1, num2, sum;

// Get input from user

printf("Enter first number: ");

scanf("%d", &num1);

printf("Enter second number: ");

scanf("%d", &num2);

// Add the numbers

sum = num1 + num2;

// Display the result

printf("Sum: %d\n", sum);

return 0;

}

The compilation process involves pre-processing, compiling, assembling, and linking, resulting in an executable file that can be run on the target machine.

1. **Explain the differences between a compiler and interpreter.**

* In compiler the compiled codes run faster than interpreter while interpreted codes run slower than compiled codes.
* The compiler saves the machine language in form of machine code on disks while the interpreter does not save the machine language.
* The compiler generate output in the form of (.exe.) while interpreter does not generate any output.
* The compiler can see code upfront while the interpreter works by line working of code.
* Compiler does not require source code for later execution while interpreter requires the source code for later execution.
* In compiler object code is permanently saved for future use while in interpreter no object code is saved for future use.

1. **List all main categories available in c programming.**

Arithmetic Operators.

Relational Operators.

Logical Operators.

Bitwise Operators.

Assignment Operators.

Other Operators.