Define

1.Compiler:software tool that translates high level source code written in a programming language into machine code

2.Source code:Refers to human readable code written by a programmer in a high-level programming llanguage

3.object code: Output of the compiler consisting of machine code or intermidiate code that is not human-readable

4.Linkers: Tools thaat compine object code files and libraries to produce an executable program.

Difference between a compiler and interpretor.

Compiler translates entire source code to object code while interpreter translates the source code to object code bit by bit at a time that is line by line.

Compilation process happen before execution i.e one compiled, the program can run multiple times without needing recompilation while interpretation occures during runtime.

Compiler often catch syntax errors and some semantic errors in the code during the compilation process while interpretor detects error on line, tge interpretor stops the execution of the program and provides an error message.

Compiled code is usually faster in terms of execution because it's directly translated into machine code while interpreted code tends to be slower because it's translated and executed line by line which incures some overhead.

To run compiled program on different platform l, rhe source code needs to be recompiled for each platform while interpreted code is generally more portable as long as there is an interpreter availed for the target platform.

Compiled program often require less memory because they are optimized during the compilation process while ihterpreted programs might consume more memory as interpretor needs to be load in memory along wuth the source code.

List all main categories of operation available in c programming and the specific operators under each category.

Arithmetic oooperators

+(addition)

-(subtraction)

\*(multiplication)

/(division)

%(modulus, gives the remainder of division)

2. Relational operators

==(equal to)

!=(not equal to)

<(less than)

>(greator than)

<=(less than or equal to)

>=(greater than or equal to)

3. Assignment oooperators

•=(assignment)

+=(addition assignment)

-=( subtraction assignment)

\*=(multiplication assignment)

/=(division assignment)

%=( modulus assignment)

&=(bitwise assignment)

|=(bitwise or assignments)

^=(bitwise xor assignment)

<<=(left shift assignment)

>>=(right shift assignment)

4. Logical operator.

&&(logical AND)

`||`(logical OR)

`!`(logical NOT)

5. Increment and decrement operator

++(increment by 1)

--(decrement by 1)

6.Bitwise operators

&(bitwise AND)

|(bitwise OR)

^(bitwise XOR)

~(bitwise NOT)

<<(left shift)

>>(right shift)

7. Conditional( ternary) operator

?:( conditional operator, used for decision making.)