-> Compiler-construction: a compiler is a computer program that translates a program written in one language (source code) into an equivalent program (target lang) in another lang.

input

in the form of high level lang, like, c, c++, java

input ——→ [compiler] ——— Target program

↳ can be IR

means Intermediate code Representation or Assemble language.

-> compiler construction: The process of constructing a ~~computer~~ compiler is called compiler construction.

-> why we construct compiler: without compilers, our machines (computers) are not able to understand the source code, written in high level lang.

-> Types of compilers:

1) Single pass

it compile the whole process in only one-pass.

ex pascal

Source code [compiler] → Target code

Means it does not generat the IR

2) Multi Pass:

it compiles the process/source code of a program Multiple times.

ex java.

Source code [Front End] IR [Middle end] IR [Back end]

→ cousins of compilers: The phases of compiler constsuction is known as Phases cousins of compilers and that process is also know as language processing system.
Phases

→ 1st cousin:

Pre processor:-

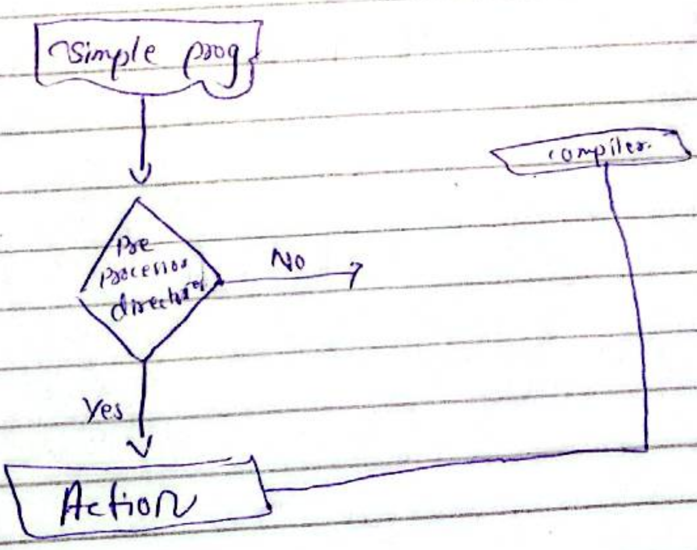
The pre-processor accepts source code as input and is responsible forms:

1, → it removing comments:

2, Process all pre processor directires denoted by #.

means what ever # symbols will be included in our code, it will replace it will original coding.

→ diagram.



Machine code

then that code comes as an input to the compiler.

② Compiler: it takes pure HLL as input and convert it into assemble code.

③ assembler: it takes as a input assemble code and convert it into the relocated machine code.

④ Linking and loading:
it perform the four functions:

① Allocations: The relocatable machine code space will acquired and in memory that spaced will be assigned

② Relocation: All tasks will be load in main memory. from secondary to main memory. whose tasks periority is high.

3 Liker: it is a tool used to link all parts of program togather for execution into a single executable file.

④ Loader: A loader loads this executable file into the memory and starts its execution.

⇒ Phases of compilers.

There are six Phases involve to construct the compiler.

-> source code. After removing directives the
→ code comes in the first phase of compiler as an input to the ~~th~~ first Phase of compiler.

1 Lexical Analysis. it collects all the tokens an generate the output as stream of tokens. and these stream of tokens goes to second phase.

2, Syntax Analysis Phase (error reporting) :- parse tree generate as a output and it will go to semitic Analaysis Phase:-

3 Semintic Analysis phase (error reporting) then it will convert into the syntax tree. 4phase.

4 Syntax tree. then

5, IR (Intermediate code generation.

6 IR (Intermediate ~~code generation~~ Representation)

7 code optimizer

8, code optimized

9 Target code generation.