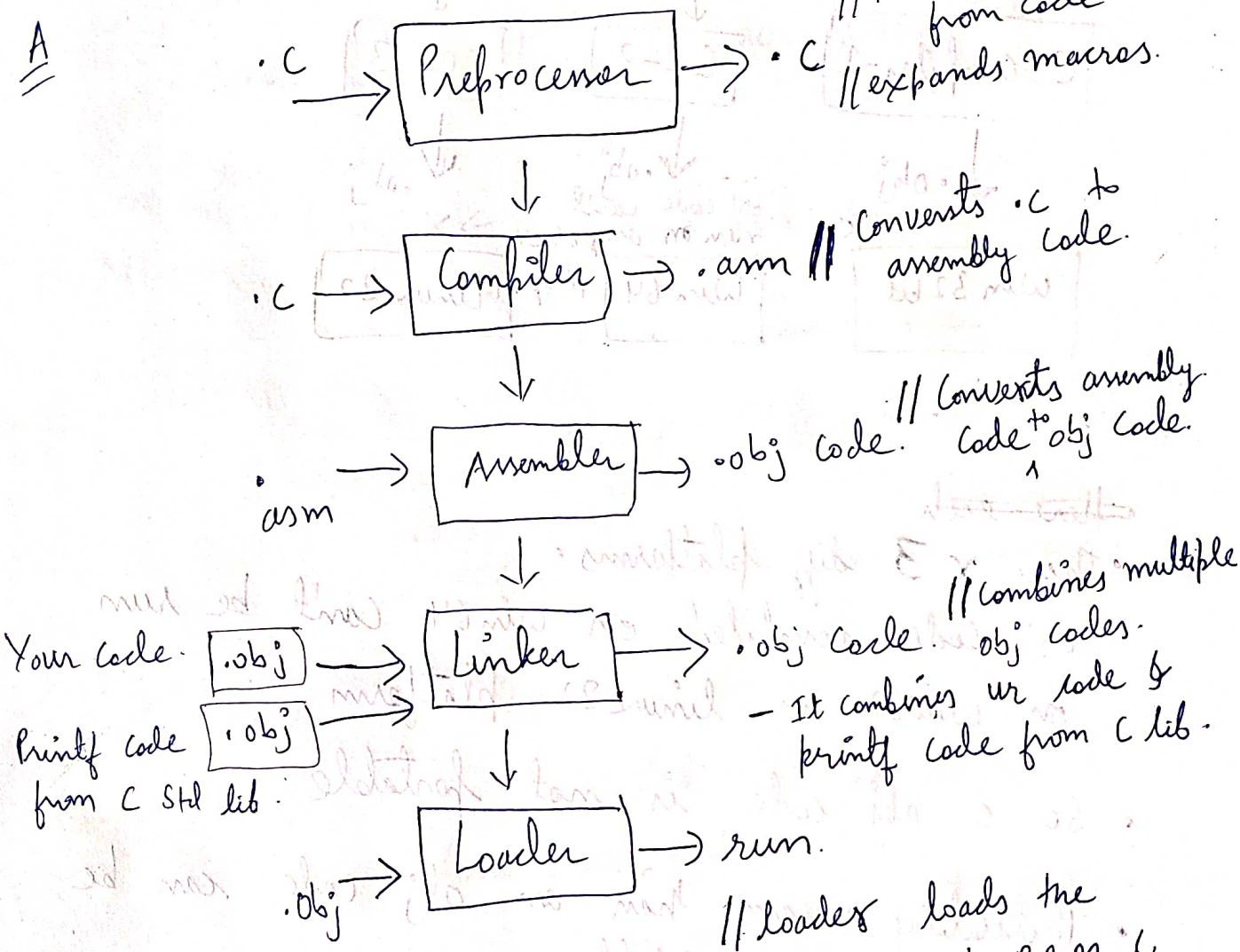
Introduction to Java

Unit-1

Q) Phases in the execution of a C program / Lifecycle of a program.

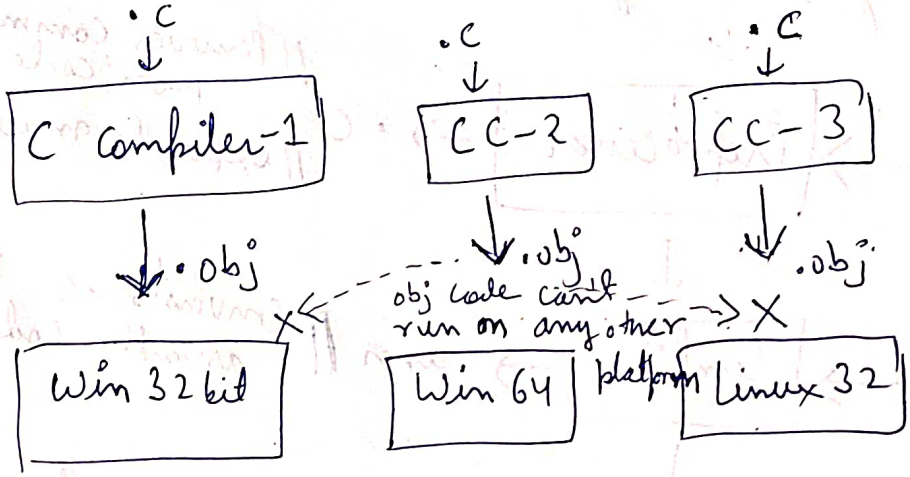
A



- obj code is in binary format.
- obj code is "a.out" in linux
  "a.exe" in win.

Q) C obj code is not portable.

- obj code of a particular ~~format~~ system can run only in the particular system it is compiled
- It can't be run on any other system.



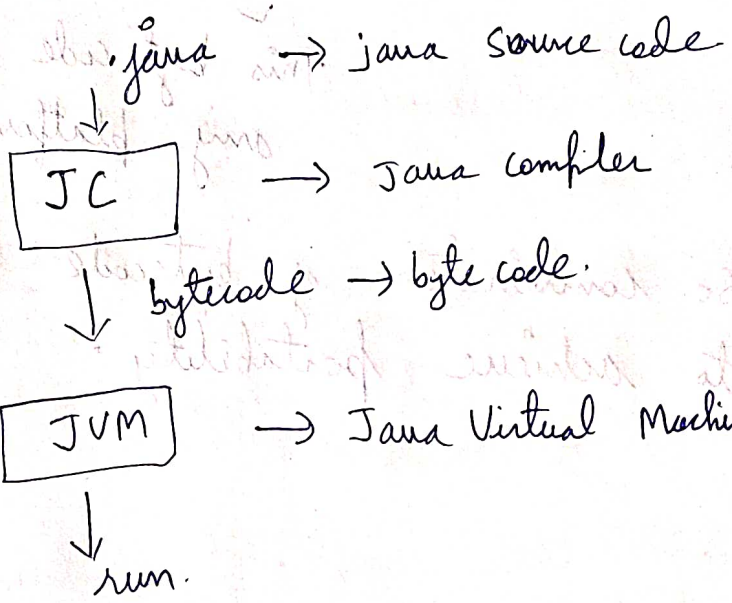
~~Here each~~

- There r 3 diff platforms.
  obj code compiled on win64 can't be run on win32 or linux32 platform
- So c obj code is not portable.
- Portability means than an obj code can be run on any platform.

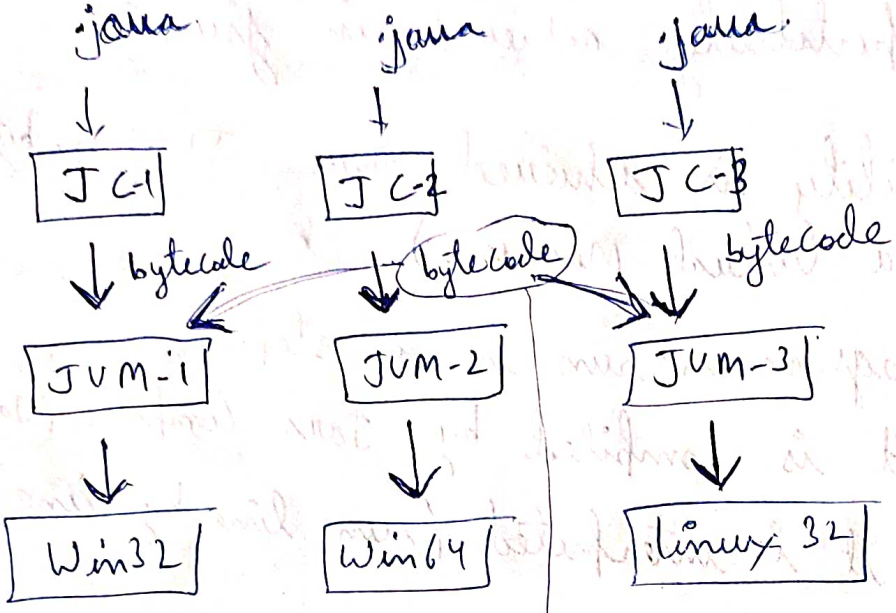
Q) How is portability achieved in Java?

A) • ~~For~~ Portability is achieved thru JVM & bytecode (Java Virtual Machine).

• Java program is run in 2 steps
  1) 1st it is compiled by Java Compiler (JavaC)
  2) Then, it is interpreted (run line by line) by JVM.



• When java code is compiled, it is converted to a bytecode (not obj code)
• This bytecode can be run on any systems/platforms
• So bytecode is ~~portable~~ portable.



This bytecode can be run on any platform using JVM.

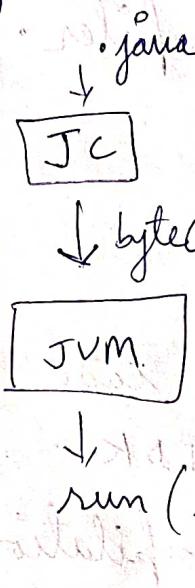
- So combination of bytecode & JVM helps to achieve portability.

Q) Is java compiled or interpreted?

A) Java is both -

first, it is compiled to bytecode by java compiler

then, ~~it~~ ~~is~~ this bytecode is interpreted line by line by the JVM.



- Bytecode is an intermediate form (similar to obj code)
- It is an highly optimised set of instructions wch. is interpreted by the JVM.
- ~~Same~~ bytecode is portable - it can be run on JVM in any system/platform

Q) Write short notes on :-

1) Java Compiler
2) Java Virtual machine
3) Bytecode
4) JDK (Java development Kit)
5) JRE (Java Runtime Environment)
6) Just in time compiler.

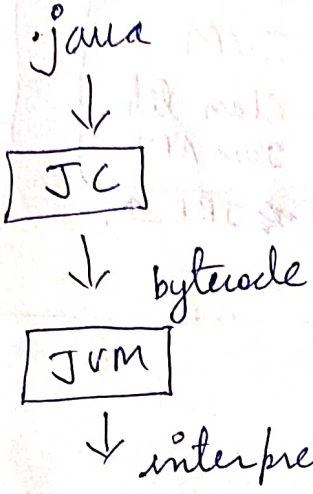
---

1) Java Compiler

- Converts java source code to bytecode
- It is a part of JDK.
- It also detects compilation error in the code.

2) JVM

- It is an interpreter which runs the bytecode line by line.
- JVM helps in achieving portability
- JVM can run bytecode compiled in any system.
- JVM is a part of JRE.

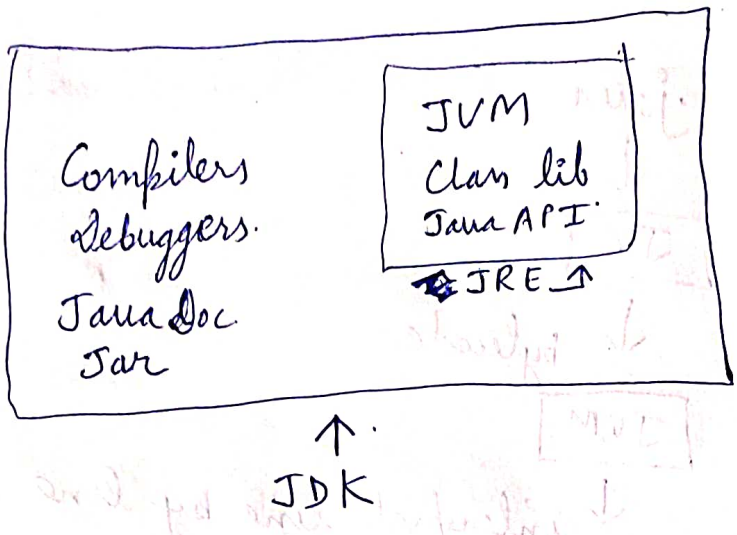


3) Bytecode

- It is ~~an~~ highly optimised set of instructions
- JC converts .java to bytecode
- Bytecode obtained in one system/platform can be run on any other platform.
- B/C of JVM helps in achieving portability.

4) JDK (Java Development kit)

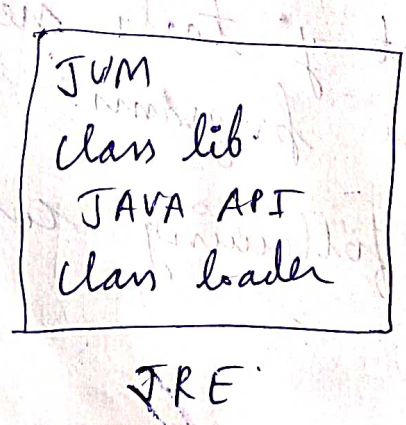
- To run java programs & develop s/w JDK is needed.
- It is a set of tools needed to develop & run java programs.
- JDK has following components.



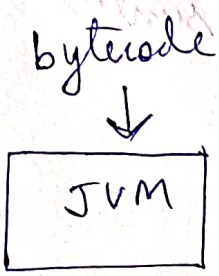
- JDK consists of JRE & other tools such as compiler, debugger, Javadoc etc.
- JRE contains JVM & other tools to run java programs.

5) JRE

- Java Runtime Enviroment is a part of JDK.
- It contains several tools for execution of java program.
- JVM is also a part of JRE.

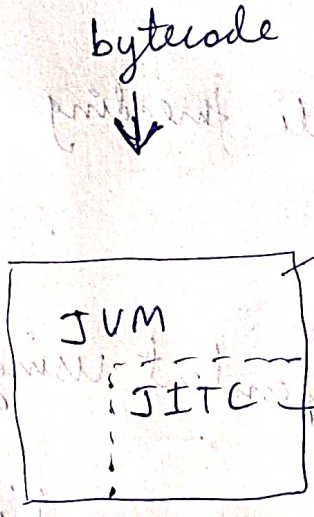


6) Just-in-Time Compiler (JITC):



- JVM interprets b/c line by line
- This causes some slowdown as interpretation is slower than compilation

- To ~~resto~~ make it fast - a JITC is added to the JVM.
- This JITC can run multiple lines at a time
- However, only selected part of the code is ~~rate~~ run multiple lines at a time by the JITC
- JITC can't run the complete code.



- It is a part of JVM.

Features of java:

1) Syntax similar to C & C++
   - for, while, if else use same syntax

2) Robust & Reliable
   - Java is less prone to errors.
   - Because it is strictly typed, wh. reduces compile time / syntax error.
   - It has exception handling feat. to reduce runtime errors.
   - Garbage Collector to handle m/m

3) Object Oriented
   - It is purely OO
   - Everything is inside class.
   - Provides oops features - inheritance, encapsulation, poly etc.

4) Multithreading support
   - In-built support for multi-threading
   - improves speed

5) Portable
   - Java is platform independent using JVM & bytecode
   - Same b/code can be run on any platforms

6) Java is interpreted still it is fast
- JVM interprets line by line which makes it slower.
- However, it also has Just in time Compiler wh. can run mul lines at a time
- It improves speed.

7) Distributed
- J can be used in a distributed environment over the internet.

8) Secure
- J is secure as it perfoms bounds checking in arays.
- JVM is isolated from other parts of computer, so it keeps the computer secure.

9) Dynamic
- J supports strings, obj & arays thru dy. m/m allocation.
- static obj, ar, strings not allowed.