M T W T F S S

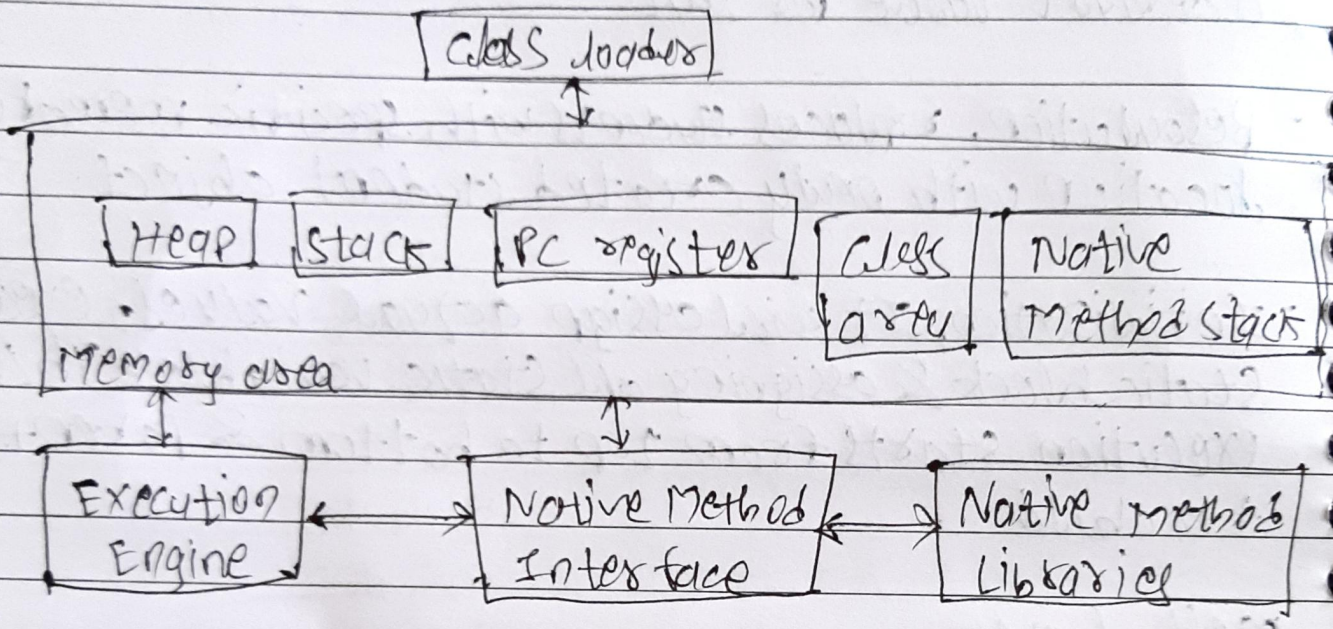
Page No.: YOUVA

Date:

Assignment-2

Q.1) Explain the components of the JVM.

→ There are 5 major components inside JVM.



- Class loader -

- It has 3 major responsibilities. Loading, Linking, Initialization.
- Main responsibility is taking a class & loading into the memory (ram) for execution from secondary memory.
- It is responsible for reading .class file, generates binary data saving it in method area.

- Linking is divided in 3 parts - verification, preparation, Resolution.
- In verification, it will check whether the class file is safe to execute or not.

- In preparation, If you use an instance level variable or static variable in your class. it assign a default value for that.

- Relocation, replaces students with specific memory location with newly created student object.

- Initialization, It will assign actual values, execute static block & assigning all static variables. This execution starts from top to bottom & parent to child.

Method area -
- It has 5 parts.
- method area / Heap area -
  - Load all class information JVM has only one method area & heap area.
  - It is also called as thread memory.

- Stacks :-
  - Keep method information, local variables.
  - A separate runtime stack is created for every thread.
  - All details are stored here until completion of method.

- Pc Register -
  - It holds information about next execution.
  - It stores address of currently executing JVM Instruction.
  - Separate Pc register is created for every thread.

- Native method stack -
  - Thread creates this kind of memory & thread is at a whole new level.

- Native Method Area -
  - This is stack that can support native methods that are written in different language.

- Execution Engine -
  - It is responsible ~~for~~ to run particular program.
  - It is divided in 3 parts.
  - It is responsive in executing byte code by converting it into machine code. & also interact with memory area.
  - Interpreter -
    - Responsible to read, interpret & execute java program line by line.
  - JIT compiler -
    - Main job is to overcome interpreter disadvantage of slowness during execution.

- It works only for repeated methods not single method.
- By lowering overall compilation time it increases performance.

- Garbage Collector -
- Used to free up memory by collecting & removing objects from heap area when there is no reference

- Native method libraries -
- Contains libraries required for execution engine.

- Native method interface (JNI) -
- Acts as bridge between Execution Engine & Native method library.
- It also provide non-programming language packages allowing developer to write code in various language.

Q.2

JDK	JRE	JVM
1) Java Development Kit	Java Runtime Enviroment	Java Virtual Machine.

JDK -
- Java Development Kit, an environment of software development used for developing applications.
- It consists JRE & development tools.
- It has 3 editions.
  - Java SE
  - Java EE
  - Java ME
- It has JVM, loader, interpreter, compiler, Javadoc, jar.

JRE -
- Java Runtime Environment or Java RTE
- Set of tools designed for running softwares
- Implementation JVM & JRE provides runtime environment.
- User need JRE to run any java program

JVM -
- Java virtual machine, provide enviroment for java applications & code.

M T W T F S S

Page No:

Date:

YOUVA

- Converts Java byte code to machine language.
- It is capable of running programs written in other languages, compiled in java bytecode.
- It doesn't exists physically.
- It is essential part of JRE.
- It perform following task.
  - Provides Runtime Environment
  - Verified code
  - Loads code
  - Executes code