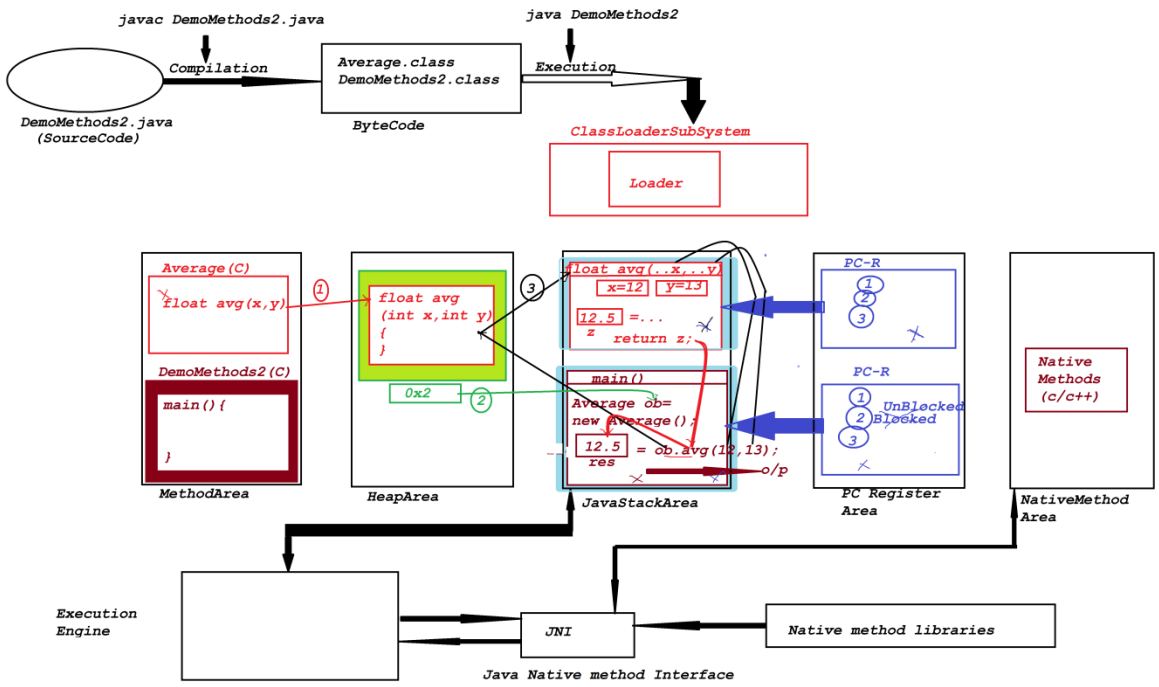
Dt : 24/7/2023

Execution flow of DemoMethods2.java:

ClassFiles:

Average.class

DemoMethods2.class(MainClass)



==========

(JVM...cont...)

(d)PC Register Area:

=>Program Counter(PC) registers will record the status of method execution in JavaStackArea.

=>Every method which is executing in JavaStackArea will have its own PC-Register

and all these PC-Registers are opened in a separate memory block known as

PC-Register Area.

=>These Program Conter Registers will control and moniter the execution process

of ExecutionEngine when shifted from one method to another method.

==========

(e)Native Method Area:

=>The methods from JavaLib which are declared with "native" keyword are known as

Native methods.

=>These native methods internalling having c/c++ codes.

=>when these native methods are used in applications then ClassLaoderSubSystem

will separate these methods and load on to a separate memory location known

as Native method Area.

=>ExecutionEngine will use JNI(Java Native method Interface),a internal component

to execute Native methods.

==========

3.Execution Engine:

=>ExecutionEngine is a processor or executor of JVM and which starts the execution

process with main() method available in JavaStackArea.

=>ExecutionEngine internally having two translators:

(i)Interpreter

(ii)JIT-Compiler

(i)Interpreter:

=>Interpreter will start the execution process and executes normal Instructions.

=>when interpreter finds stream-instructions(Bulk-data-transfer) then it transfers the control to JIT-Compiler.

(ii)JIT-Compiler:

=>JIT(Just-In-Time) Compiler will execute Stream Instructions or Multi-Media Instructions.

==========

faq:

why Javalang use Interpreter in execution-process?

=>when we have interpreter in execution-process,then we can accept the request

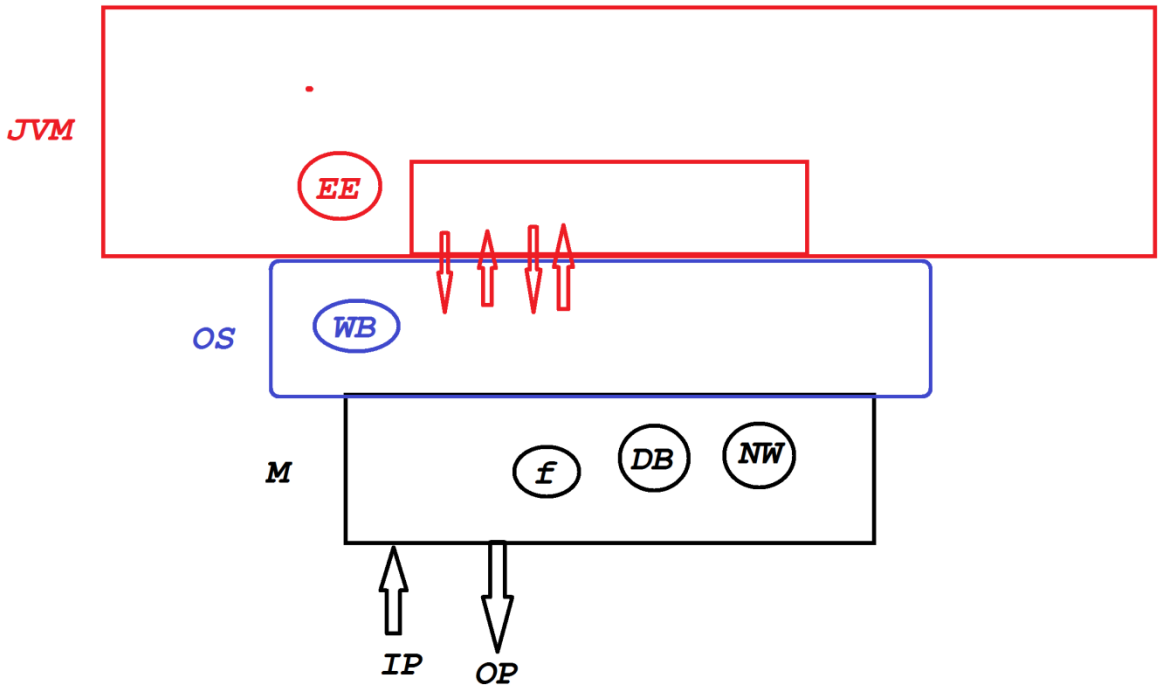
in the middle of execution-process and which is preferable for client-server

application development.

==========

Note:

=>Java is interpreted language beacause it uses interpreter in execution process.



==========