DD MM YYYY
12 01 2024

Module - 1

2(a). • The fundamental idea behind the Virtual machine is to abstract a single hardware into several different execution environment, thereby creating the illusion that each execution environment is running its own private computer.

• create the illusion that the process has its own processor and its own memory.

• The Host OS will be the main OS. and the installed OS are called as guest OS.

• First time (operating machine) virtual machine appeared as virtual machine operating System for IBM mainframes in 1972.

Processor	Processor	Processor
Kernel	Kernel	Kernel
VM_1	VM_2	VM_3
virtual machine implementation		
Hardware		

System mode: virtual machine mode.

*Benifits :→

• V.M. is Able to. Share Same hardware and run different execution environment.

D D M M Y Y Y Y

- Host System will be protected from guest OS and each virtual machine is protected from each other.
- A virus in Guest (OS) (virtual machine) will not affect the other guest virtual machine and host System.
- Event though virtual machines are Seperated and the Software resource is shared among them. (VM).
- In virtual machine, the problem like System development time is eliminated.
- The user programs are executed on a virtual machine and System development is done in other environment
- V. Machine helps in rapid testing of user code in different environments.
- System insolidation: two/more System made to run in Single System.
- eg: Java virtual machine.

In

- Java virtual machine the Source code is compiled by .class files into byte code. Byte code is the binary information that will be run on Java virtual machine.
- Java virtual machine consists of a class loader & JAVA

D D M M Y Y Y Y

interpreter. The classes are loaded from both the Java Program and java API.

- The class loder loads compiled .class files from both java program and java API. and for the execution of java interpreter.

- Then it checks the .class files for validity.

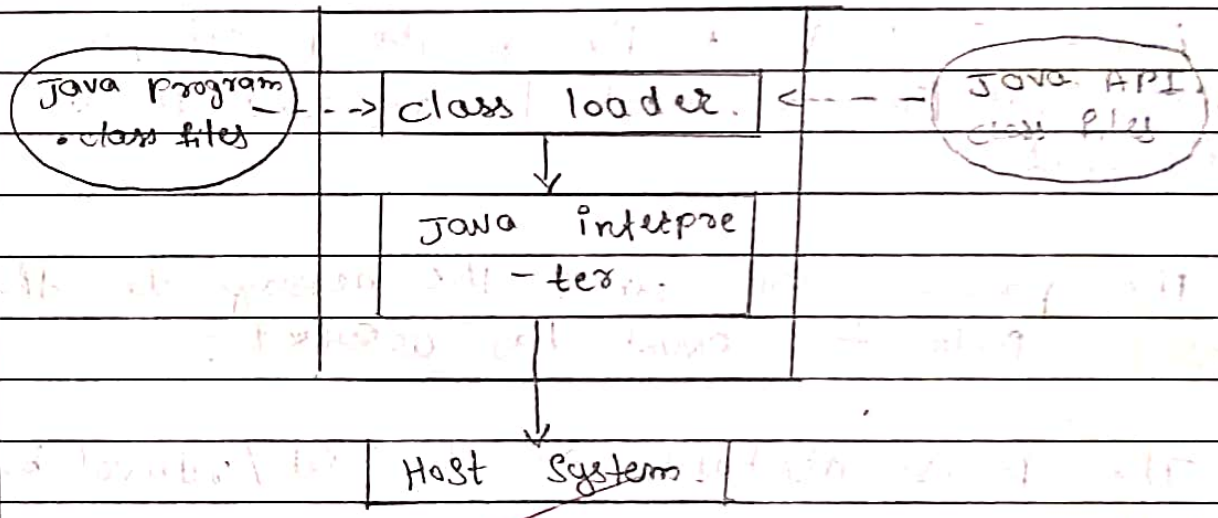


fig: The Java virtual machine.

(b) • System call is the call by the controler to perform the particular (specified) function / operation.

- There are 6 major types of System calls are these
  (i) Processor control
  (ii) file management
  (iii) Device management.
  (iv) Information managment
  (v) Protection
  (vi) communication.

cont...

b → (i) Process Control :→

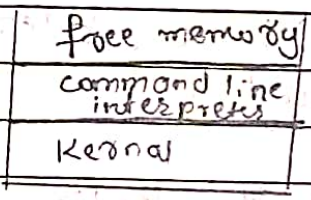
- The Process control System call includes end, load, abort, launch, create a Process / delete a process, get / set Process attribut, wait for time, wait for event, Signal event etc.

- The process must be created, launched and terminated. resumed and eventually stopped.

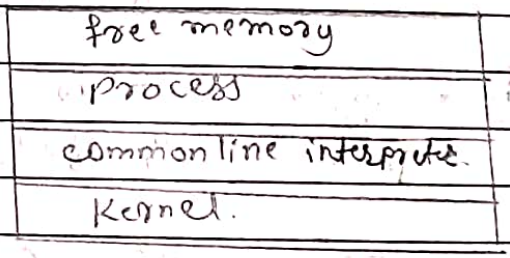
- After creating a Process the parent process may have to wait (for time) or wait for an event to occur (wait for event).

- The process will Sends the message to the parent process After the event has occured.

- The Process attributs can be set / retrived by using System calls.



(a) Startup of System.



(b) running a Process.

(ii) File management :→ • The file management System call includes create / delete a file, get / set file attributes, read, write, open, close etc commands.

D D M M Y Y Y Y

- After creating a file, file is open and data is read / written into file.

- The file attributes like file name, file type, permission etc can be set / retrived by using system calls.

- These operations also supports for a directory / ordinary files.

* Device management :→

- The Device management System call includes the request for device, logically atach / detach the device and release device, get / set device attribute etc.

- If (When) a process and needs a resource it has to send a request for device, then the control granted the process to use resources.

- In multiprogramming systems ~~the proo~~ once the process has used a resource it has to release that / return that resource to the OS. So that other process can use the resource.

- The Device may be physical (disk drives) or virtual (RAM disks, files).

D D M M Y Y Y Y

* Information management :-

- Information management System call includes the calls like file / process / device attributes, current time, date etc
- It is responsible for transfering data between the OS and the user
- The information like no. current users, version of OS, current time, data, data system can be transferred from OS to users by using these system calls.

* Communication :-

- This System call includes IPC (inter process communication) message passing and Shared memory.
- Message passing is simpler / easier, has system calls for read & write the processes, applicable when we have small amount of data.
- Shared memory :- faster
  - Applicable when we have large amount of data, difficult to implement,
  - has few System calls.

* Protection :- This System call provides the mechanism for controlling which process

D D M M Y Y Y Y

User can access which resource

Module - 2

4. (a)

FCFS :-

Process	AT	BT	CT	TAT	WT
P_1	0	9	12	12	3
P_2	1	4	15	14	10
P_3	2	9	23	21	12
P_4	3	5	27	24	19

Gantt chart

P_1	P_2	P_3	P_4	P_1	P_2	P_3	P_4
0	1	2	3	4	12	15	23	27

average T.A.T = 17.75 ms.

average WT. = 11 ms.

ATAT
average Turn around time.

* STRF :-

Process	AT	BT	CT	TAT	WT
P_1	0	9	18	18	9
P_2	1	4	5	4	0
P_3	2	9	27	25	15
P_4	3	5	10	7	2

Gantt chart

A	P_2	P_2	P_2	P_2	P_4	P_1	P_3
0	1	2	3	4	5	10	18	27

D D M M Y Y Y Y

average waiting time = 6.5 ms.

average T.A.T = 13.5 ms.

* Round Robin q = 2 ms.

Process	A.T	B.T.	C.T.
P_1	0	~~9~~ 7 5	
P_2	1	~~4~~ 2	
P_3	2	~~9~~ 7 5	
P_4	3	5	

Ready queue: ~~P_1~~ | P_2 | ~~P_3~~ | ~~P_4~~ | ~~P_1~~ | ~~P_2~~ | P_3 | ~~P_4~~ | P_3 |
0

Running queue: | P_1 | P_2 | ~~P_3~~ | P_1 | P_4 | P_2 | P_3 | P_4 | P_3 | P_4 | P_3 |
0 2 4 6 8 10 12 14 16 18 19 21

R.Q. | ~~P_1~~ | ~~P_2~~ | ~~P_3~~ | ~~P_1~~ | ~~P_4~~ | ~~P_3~~ | ~~P_1~~ | P_3 | P_1 |
0

| P_1 | P_2 | P_3 | P_1 | P_4 | P_3 | P_1 | | |
2 4 6 8 10 12 14

Quiz 2

D D M M Y Y Y Y

(1) (b) create ✗

(2) (a) ✓

(3) (b) ✓

(4) (b) ✓

(5) (a) 4 ✗

m-2

4(c) (a) Priority

	A.T	B.T	Priority	C.T	T.AT	W.T.
P_1	0	9	3	22	22	13
P_2	1	4	2	14	13	9
P_3	2	9	1	11	9	0
P_4	3	5	4	27	24	19.

Grantt chart

P_1	P_2	P_3	P_2	P_1	P_4
0	1	2	11	14	22	27

average TAT = 17 ms. ✓

average W.T = 10.25 ms. ✓

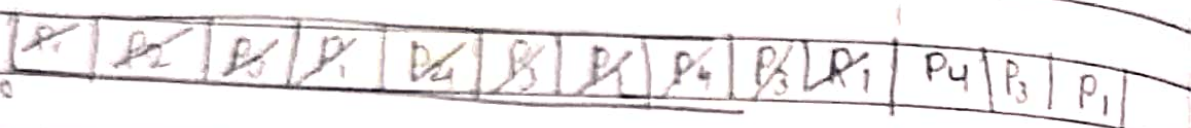
D D M M Y Y Y Y

AM-2

4(a) R.R

Priority	Process	A.T	B.T	CT
3	P_1	0	9	23
2	P_2	1	4	
1	P_3	2	9	
4	P_4	3	5	

Ready



Running