VIRTUAL MACHINE

Virtual machines are software's on which other software's can be executed as they are executing on a physical machine.

There are two types of virtual machine:

1) Application/ process virtual machine.

2) System/hardware virtual machine.

1) Application/process virtual machine: In this we can take example of JVM. Because of JVM, Java is a platform independent language.

2) System/hardware virtual machine: In this we can take example of Virtual Box. Which gives as ability to run multiple of operating systems on a single physical machine. Here operating system is also a programming language which should be supported by Virtual Box. Some advantages of Virtual Machine:

- Allows use of multiple operating system on a single physical computer interdependently.
- Easy to manage and maintain.

Some drawbacks of Virtual Machine:

- Not as efficient as a physical computer.
- Multiple virtual machines running on a single physical machine can deliver unstable performance.

Viva Vice on Virtual Machine

Q1. What is Virtualization?

The process of creating virtual versions of physical components i.e., Servers, Storage Devices, Network Devices on a physical host is called virtualization.

Q2. Mention what are the different types of virtualization available?

Application virtualization, Presentation virtualization, Network virtualization and Storage virtualization.

Q3. What is virtual networking?

A network of VMs running on a physical server that is connected logically with each other.

Q4. What is vSS?

vSS stands for Virtual Standard Switch which works like a physical switch, automatically detects a VM which want to communicate with other VM. It is responsible for communication of VMs hosted on a single physical host.

Q5. What is vDS?

vDS stands for Virtual Distributed Switch acts as a single switch in a whole virtual environment and is responsible to provide central provisioning, administration, and monitoring of the virtual network.

Q6. What is datastore?

Datastore is a storage location where virtual machine files are stored and accessed.

Q7. What is the .vmx file?

It is the configuration file of a Virtual Machine.

Q8. What information .nvram file stores?

It stores BIOS related information of a VM.

MCQs on Virtual Machine

Q1. Which of the following is another name for system virtual machine ?

a) hardware virtual machine

b) software virtual machine

c) None of the mentioned

Q2. Which of the following provide system resource access to virtual machines ?

a) VMM

b) VMC

c) All of the mentioned

Q3. Your colleague has accidentally allocated more vRAM than your company are licensed for. What will happen to your virtual machines?

a) All VM's will be Powered Off

b) New VM's can not be Powered On

c) Nothing will happen

Q4. Point out the wrong statement :

a) Load balancing virtualizes systems and resources by mapping a logical address to a physical address

b) Multiple instances of various Google applications are running on different hosts

c) Google uses hardware virtualization

Q5. Point out the correct statement :

a) A virtual machine is a computer that is walled off from the physical computer that the

virtual machine

is running on

b) Virtual machines provide the capability of running multiple machine instances, each with their own operating system

c) All of the mentioned

Q6. A memory reservation defines:

a) the amount of virtual machine memory that can be paged to the VM kernel swapfile

b) the amount of physical memory that is guaranteed to the VM

c) he amount of host memory reserved for the VMkernel

Q7. Which of the following formulas defines the amount of virtual machine memory that will always be composed of disk pages?

A. Memory allocated -(minus) memory limit

B. Memory limit -(minus) memory reservation

C. Memory allocated -(minus) memory available

Q8. For paying an extra level of indirection for each memory access, Virtual machine monitor maintains

a)Shadow page table

b)Stack table

c)Memory stack

Q9. Two processors running one is user process, other is operating system process, latter is called

a)Kernel process

b)Supervisor process

c)both a and b

Q10. Software that supports Virtual machines, is called

a)Virtual machine monitor

b)Hypervisor

c)both a and b

MCQs Answers

- Q1. (a)
- Q2. (a)
- Q3. (b)
- Q4. (c)
- Q5. (c)
- Q6. (c)
- Q7. (c)
- Q8. (a)
- Q9. (c)
- Q10. (c)

Principles of Programming Languages:

EasyExamNotes.com covered following topics in these notes.

- Language Evaluation Criteria
- Influences on Language Design
- Language Categories
- Programming Paradigms
- Compilation
- Virtual Machines
- Programming Environments
- Issues in Language Translation
- Parse Tree
- Pointer and Reference type
- Concept of Binding
- Type Checking
- Strong typing
- Sequence control with Expression
- Exception Handling
- Subprograms
- Fundamentals of sub-programs
- Scope and lifetime of variable
- static and dynamic scope
- Design issues of subprogram and operations
- Local referencing environments
- Parameter passing methods
- Overloaded sub-programs
- Generic sub-programs

- Design issues for functions
- co routines
- Abstract Data types
- Abstraction and encapsulation
- Static and Stack-Based Storage management
- Garbage Collection
- OOP in C++
- OOP in Java
- OOP in C#
- OOP in PHP
- Concurrency
- Semaphores
- Monitors
- Message passing
- Java threads
- C# threads
- Exception handling
- Exceptions
- Exception Propagation
- Exception handler in C++
- Exception handler in Java
- Introduction and overview of Logic programming
- Basic elements of Prolog
- Application of Logic programming
- Functional programming languages
- Introduction to 4GL

Practicals:

- Memory Implementation of 2D Array.
- Memory Implementation of 3D Array.
- Implementation of pointers in C++.
- Write a program in Java to implement exception handling.
- Write a program in C++ to implement call by value parameter passing Method.
- Write a program in C++ to implement call by reference parameter passing Method.
- Write a program in Java to implement concurrent execution of a job using threads.
- Implement Inheritance in C#.
- Implement Encapsulation in C#.
- Implement static/compiletime Polymorphism in C#.
- Implement dynamic/runtime Polymorphism in C#.

Previous years solved papers:

- PPL|RGPV|May 2018
- PPL|RGPV|June 2017

A list of Video lectures

Click here

References:

- 1. Sebesta,"Concept of programming Language", Pearson Edu
- 2. Louden, "Programming Languages: Principles & Practices",

Cengage Learning

- 3. Tucker, "Programming Languages: Principles and paradigms ", Tata McGraw –Hill.
- 4. E Horowitz, "Programming Languages", 2nd Edition, Addison Wesley

Related posts:

- 1. Sequence Control & Expression | PPL
- 2. PPL:Named Constants
- 3. Parse Tree | PPL | Prof. Jayesh Umre
- 4. Basic elements of Prolog
- 5. Loops | PPL | Prof. Jayesh Umre
- 6. Subprograms Parameter passing methods | PPL | Prof. Jayesh Umre
- 7. Programming Paradigms | PPL | Prof. Jayesh Umre
- 8. Subprograms Introduction | PPL | Prof. Jayesh Umre
- 9. Phases of Compiler | PPL | Prof. Jayesh Umre
- 10. Parse Tree | PPL
- 11. Influences on Language design | PPL | Prof. Jayesh Umre
- 12. Fundamentals of Subprograms | PPL | Prof. Jayesh Umre
- 13. Programming Paradigm
- 14. Influences on Language Design
- 15. Language Evaluation Criteria
- 16. OOP in C++ | PPL
- 17. OOP in C# | PPL
- 18. OOP in Java | PPL
- 19. PPL: Abstraction & Encapsulation

- 20. PPL: Semaphores
- 21. PPL: Introduction to 4GL
- 22. PPL: Variable Initialization
- 23. PPL: Conditional Statements
- 24. PPL: Array
- 25. PPL: Strong Typing
- 26. PPL: Coroutines
- 27. PPL: Exception Handler in C++
- 28. PPL: OOP in PHP
- 29. PPL: Character Data Type
- 30. PPL: Exceptions
- 31. PPL: Heap based storage management
- 32. PPL: Primitive Data Type
- 33. PPL: Data types
- 34. Programming Environments | PPL
- 35. Virtual Machine | PPL
- 36. PPL: Local referencing environments
- 37. Generic Subprograms
- 38. Local referencing environments | PPL | Prof. Jayesh Umre
- 39. Generic Subprograms | PPL | Prof. Jayesh Umre
- 40. PPL: Java Threads
- 41. PPL: Loops
- 42. PPL: Exception Handling
- 43. PPL: C# Threads
- 44. Pointer & Reference Type | PPL
- 45. Scope and lifetime of variable
- 46. Design issues for functions

- 47. Parameter passing methods
- 48. Fundamentals of sub-programs
- 49. Subprograms
- 50. Design issues of subprogram
- 51. Garbage Collection
- 52. Issues in Language Translation
- 53. PPL Previous years solved papers
- 54. Type Checking | PPL | Prof. Jayesh Umre
- 55. PPL RGPV May 2018 solved paper discussion| Prof. Jayesh Umre
- 56. PPL Viva Voce
- 57. PPL RGPV June 2017 Solved paper | Prof. Jayesh Umre
- 58. Concurrency
- 59. Basic elements of Prolog
- 60. Introduction and overview of Logic programming
- 61. Application of Logic programming
- 62. PPL: Influences on Language Design
- 63. Language Evaluation Criteria PPL
- 64. PPL: Sequence Control & Expression
- 65. PPL: Programming Environments
- 66. PPL: Programming Paradigm
- 67. PPL: Pointer & Reference Type
- 68. try-catch block in C++