

# **Hardware And Virtual Machine**



**Papers Dock**

---

**COMPUTER SCIENCE 9618 PAPER 3**

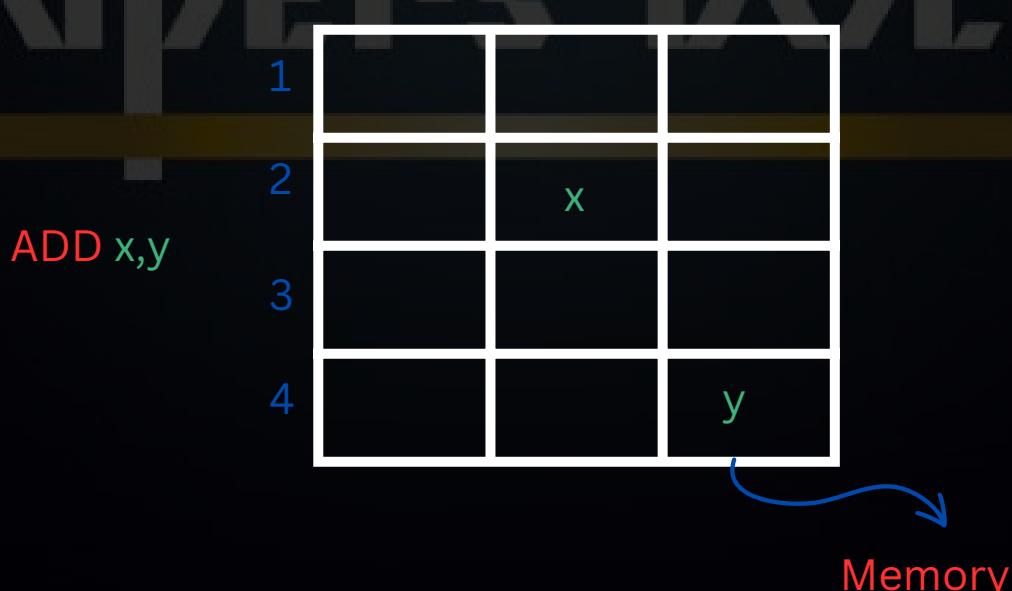
# Hardware And Virtual Machine Software

CPU architecture is the design of a computer's central processing unit. For example Von Neuman Architecture.

Every computer uses Von Neumann architecture, but advancements in computer technology have led to more complex processor designs

## Complex Instruction Set Computer (CISC)

- It is design to carry out a given task with as few lines of assembly code as possible .
- CISC processor is based on single instruction but complex which need to be converted by the processor into a number of sub-instruction to carry out the required operation .



**Question : Difference Between Fixed length instruction and Variable length instruction ?**

**Instruction 1 → Fixed Length instruction**

**Fix the amount of memory it will consume e.g 16 bits, 4 bits.**

**Instruction 2 → Variable Length instruction**

**It can take as much space as it wants.**

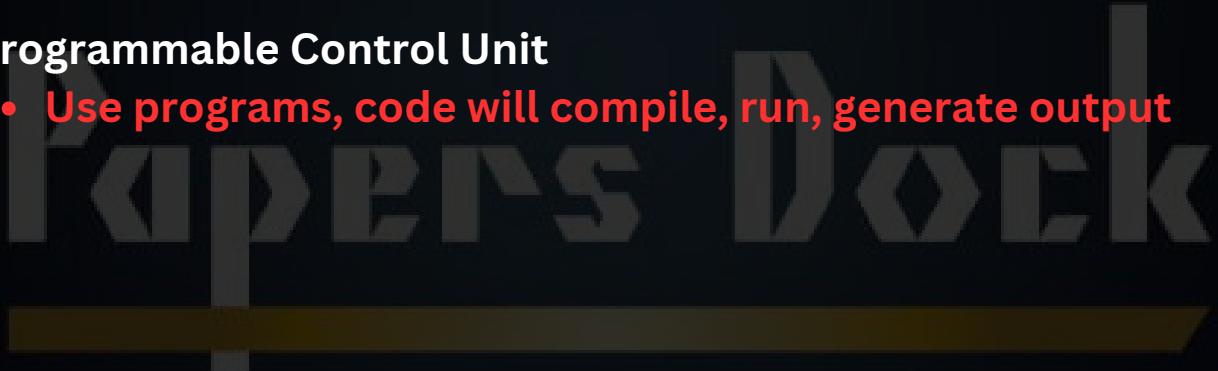
**Question : Difference between Hardwired Control Unit and Programmable Control Unit ?**

**Hardwired Control Unit**

- **Uses Logic Circuits**
- **Flip Flop**

**Programmable Control Unit**

- **Use programs, code will compile, run, generate output**



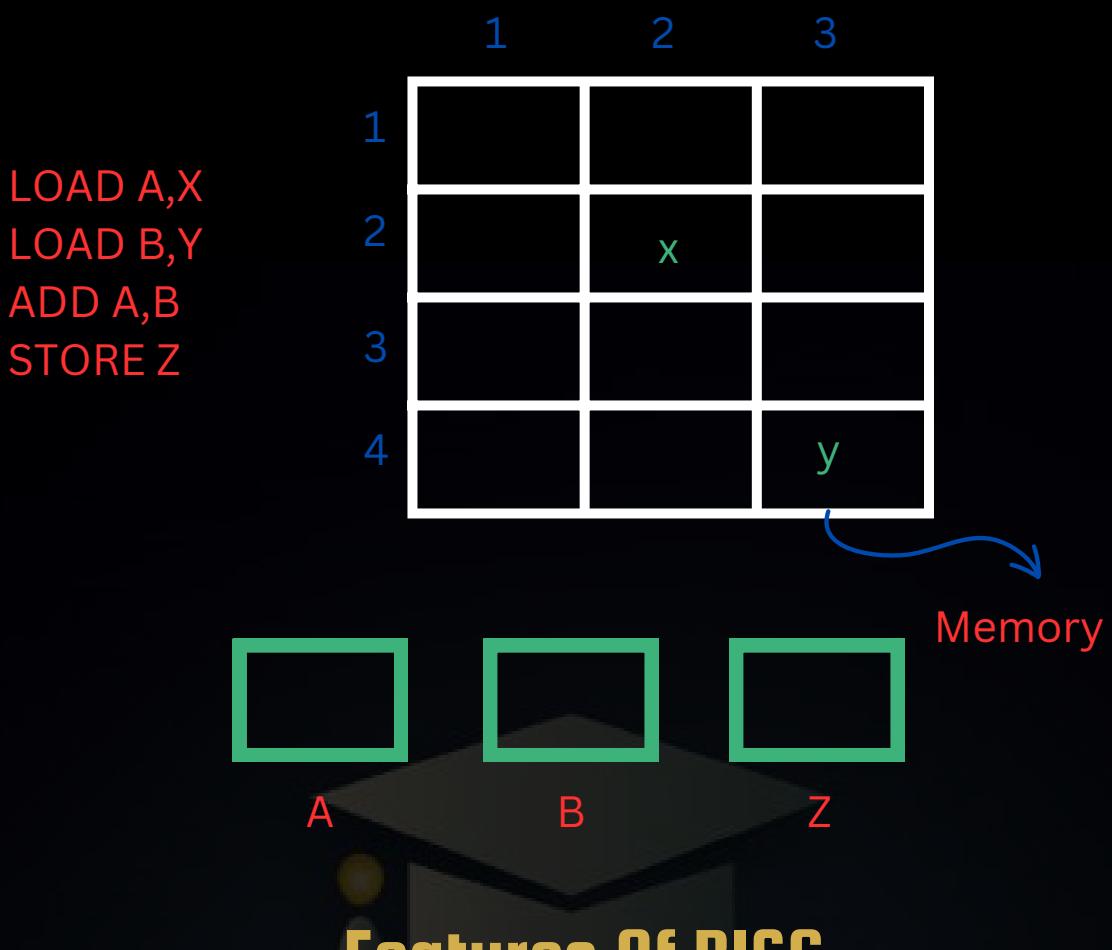
# **Features Of CISC**

- **CISC has more instructions**
- **CISC has fewer registers**
- **CISC instructions are more complex**
- **CISC has many instruction formats**
- **CISC uses multicycle instruction**
- **CISC uses variable length instruction**
- **CISC has poor pipelining**
- **CISC requires more complex circuits**
- **CISC has more addressing modes**
- **CISC makes less use of RAM**
- **CISC has programmable Control Unit**
- **CISC has many types of instructions to address memory**
- **CISC processors often include both types of control units (microcoded for complex instructions and hardwired for simpler ones)**

**Note : CISC emphasizes on Hardware ( Means more complex hardware circuits would be required to handle complex tasks )**

# **Reduced Instruction Set Computer (RISC)**

With fewer instructions, the processor achieves better performance, as it doesn't need to break down complex instructions, and Assembly Code is broken into a number of single cycle instructions.



## Features Of RISC

- Risc has fewer instructions.
- Risc has many registers
- Risc instructions are simpler
- Risc has few instruction format
- Risc usually uses single cycle instructions
- Risc uses fixed length instructions
- Risc has better pipelineability
- Risc requires less complex circuit
- Risc has fewer addressing modes
- Risc makes more use of RAM
- Risc has hardwired control unit
- Risc only uses load and store instructions to address memory
- Risc follows the Harvard architecture, which splits cache for instructions and data

Note: Emphasize on software.

## Question : What is meant by RISC And CISC Processors?

### RISC

- **Uses simple instructions**
- **Uses fixed length instructions**
- **Instructions only require one clock cycle**
- **Uses many registers**
- **Makes use of pipelining**
- **Hardwired CU**

### CISC

- **Uses many instruction formats**
- **Uses variable length instructions**
- **Makes use of different addressing modes**
- **Uses few registers**
- **Has a large instruction set**
- **Requires complex circuits**
- **Frequently uses cache**
- **Instructions (converted to sub-instructions that) may require many clock cycles**
- **Programmable CU**



# Pipelining

It is the way of improving computer performance

## Explanation

Let's assume that we are doing laundry.

One washer (takes 30 min)

One drier (takes 40 min)

One folder (takes 20 min)

It takes 90 minutes to wash, dry and fold 1 load of laundry.

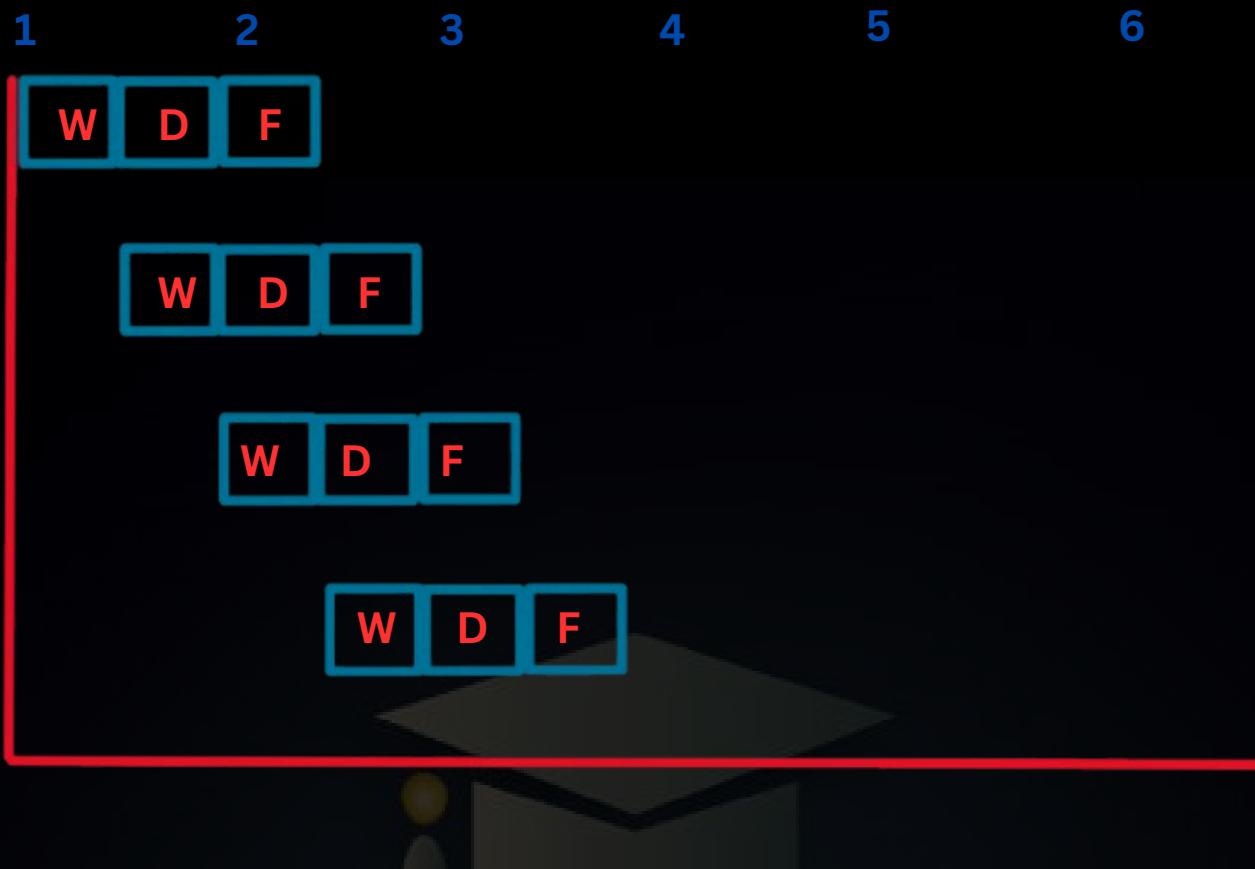
How long does 4 loads take?

## The Slow Method



If each load is done sequentially it takes 6 hours

# Laundry Pipelining



Question : What is meant by pipelining?

- Pipelining is instruction level parallelism which means multiple operations are performed in single cycle.
- Execution of an instruction is split into number of stages.
- When first stage for an instruction is completed, the first stage of the next instruction can start executing.
- Another instruction can start executing before the previous one is finished.
- Processing of a number of instructions can be done simultaneously.

## 5 Stages of Processor

1. Fetch Instruction.
2. Decode Instruction.
3. Execute Instruction.
4. Access operand in memory.
5. Write result to register.

OPCODE: What do we need to do (Decoded Instruction)

OPERAND: What do we need to do it to (Data)

There are 4 instructions ABCD

## Without Pipelining

Stage	Time interval								
	1	2	3	4	5	6	7	8	9
Fetch instruction	A					B			
Decode instruction		A					B		
Execute instruction			A					B	
Access operand in memory				A					B
Write result to register					A				

Means 1 instruction is taking 5 cycles each so if there are 4 instructions how many cycles would they need ?

# With Pipelining

Stage	Time interval								
	1	2	3	4	5	6	7	8	9
Fetch instruction	A	B	C	D					
Decode instruction		A	B	C	D				
Execute instruction			A	B	C	D			
Access operand in memory				A	B	C	D		
Write result to register					A	B	C	D	

By the help of pipelining only 8 cycles were used so How many cycles were saved ?

Question : Describe the process of pipelining during the fetch-execute cycle in RISC processors.?

- Instructions are divided into 5 stages
- Instruction fetch, Instruction decode, operand fetch, Instruction execute, write back result
- Each subtask is completed during one clock cycle
- No two instructions can execute their same stage at the same clock cycle
- The second instruction begins in the second clock cycle, while the first instruction has moved on to its second subtask.
- The third instruction begins in the third clock cycle while the first and second instructions move on to their second and third subtasks, respectively, etc.

# Interrupt Handling In CISC

- Detect Interrupt: The processor detects an interrupt signal from a device or software event.
- Save Current State: It saves the program counter and other registers to resume later.
- Identify Interrupt: The processor looks up the interrupt type in the Interrupt Vector Table to find the relevant service routine.
- Execute ISR: It jumps to the Interrupt Service Routine (ISR) to handle the specific task.
- Restore and Resume: After the ISR finishes, the processor restores the saved state and continues the main program.

# Interrupt Handling In RISC

- Interrupt Detection: When an interrupt signal is received, the RISC processor pauses its current operations to address the interrupt.
- Flushing the Pipeline: Since RISC uses a pipelined approach, it discards (or "flushes") all remaining instructions in the pipeline to prevent partial operations from affecting the interrupt handling.
- Saving State: The processor saves the current state, such as the program counter and register values, to ensure it can resume normal operations after the interrupt is handled.
- Executing the Interrupt Service Routine (ISR): The processor jumps to the appropriate Interrupt Service Routine, which is a set of instructions designed to handle the specific interrupt.
- Restoring State and Resuming: Once the interrupt is serviced, the processor restores the saved state and continues executing the main program.

**Question : Outline the process of interrupt handling as it could be applied to RISC or CISC processors?**

- Once the processor detects an interrupt at the start of the fetch-execute cycle
- The current program is temporarily stopped and the status of each register is stored on the stack.
- After the Interrupt Service Routine (ISR) has been executed
- The registers can be restored to their original status before the interrupt was detected from the stack.

**Question : Explain how pipelining affects interrupt handling for RISC processors?**

- Pipelining adds additional complexity, there could be a number of instructions still in the pipeline when the interrupt is received.
- All the instructions currently in operation are usually discarded, except for the last one the one at write-back.
- The interrupt handler routine is applied to the remaining instruction.
- Once the interrupt has been serviced, the processor can restart with the next instruction in the sequence.

# Four Basic Computer Architecture

Flynn's Classification : based on number of instructions and data



**Instruction stream :** Sequence of instructions executed by the processing unit

**Data stream :** Sequence of data or temporary result called by instruction stream

**SISD :** Single instruction, Single data

**SIMD :** Single instruction, Multiple data

**MISD :** Multiple instruction, Single data

**MIMD :** Multiple instruction, Multiple data

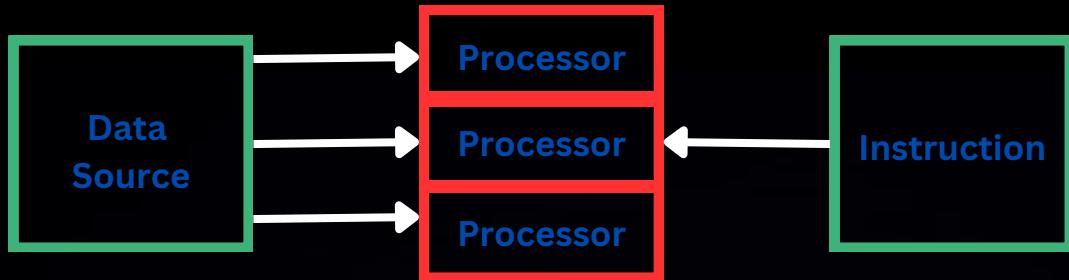
## Single Instruction Single Data



- There is only one processor, a control unit and memory unit
- The processor executes one set of instruction on one set of data.
- executes instruction sequentially

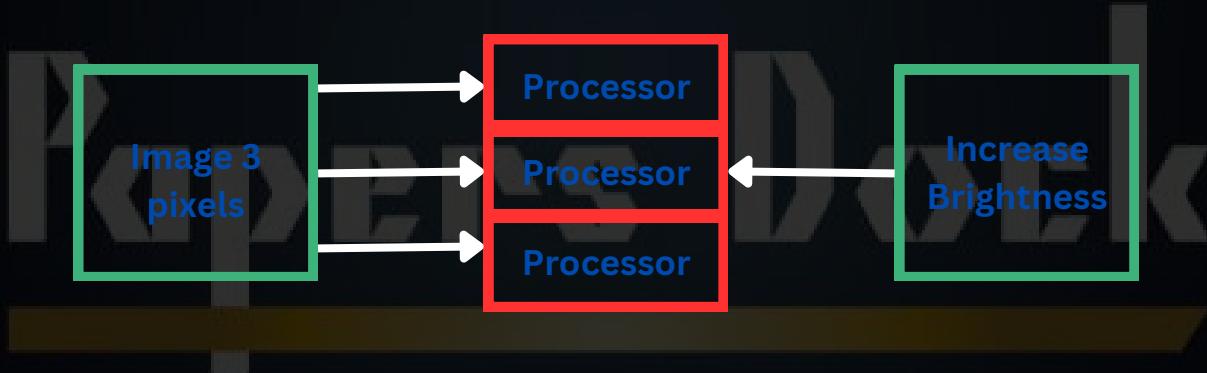
**Note :** Each task is processed in a sequential order

# Single Instruction Multiple Data



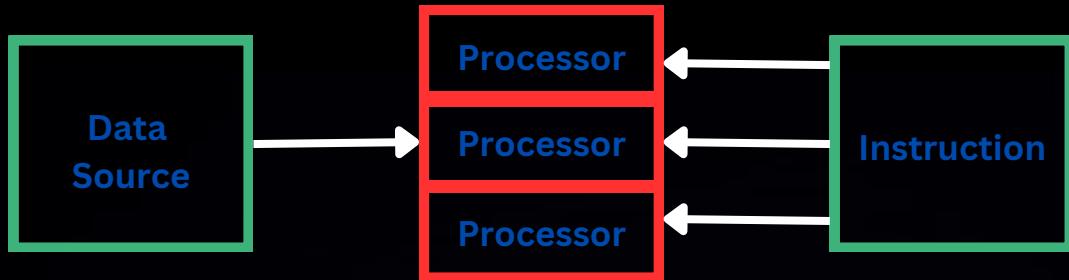
- uses many processor.
- Each processor has several ALU
- Each ALU executes the same set of instruction on different set of data at the same time.
- The instructions can be performed sequentially, taking advantage of pipelining
- Parallel computers with multiple processors

## Application



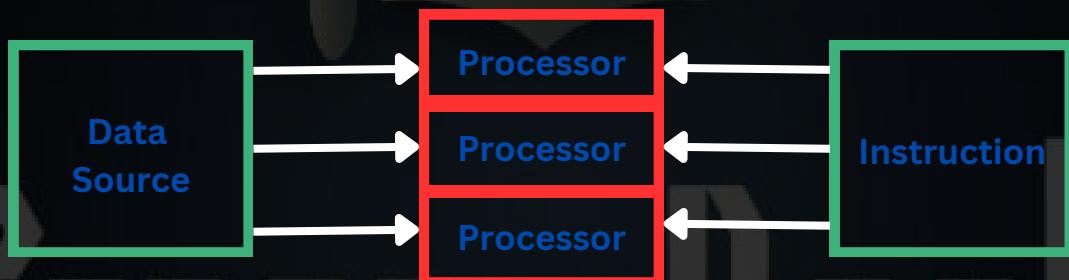
3 pixel = 3 processors  
Each processor will increase  
brightness of each pixel

## Multiple Instruction Single Data



- There are several processors
- Each processor executes different sets of instruction on one set of data at the same time.
- Parallel computers with multiple processors

## Multiple Instruction Multiple Data



- There are several processors and they work independently
- Each processor executes a different set of instruction.
- Each processor operates on different set of data.

# **Massively Parallel Computers**

**Massive :** large number of processors.

**Parallel :** to perform a set of coordinated computations simultaneously.

**Question : Describe what is meant by Massively Parallel Computers?**

- **large number of processors.**
- **working collaboratively on the same program.**
- **working together simultaneously on the same program.**
- **communicating via message interface.**



# **Issues In Massively Parallel Computers**

## **HARDWARE :**

- Processors need to be able to communicate
- So that processed data can be transferred from one processor to another.
- So it's a very challenging topology.

## **Software :**

- Appropriate programming language should be used.
- which allows data to be processed by multiple processors simultaneously.

**Question : Explain one of the hardware issues in massively parallel computers?**

- Communication between the processors is the main issue
- As each processor needs a link to every other processor
- so it's a very challenging topology.

## **Characteristics Massively Parallel Computers**

- Large number of processor
- Collaborative and simultaneous processing
- Network infrastructure
- Communicate by sending messages.

## **Conditions for massively parallel computers**

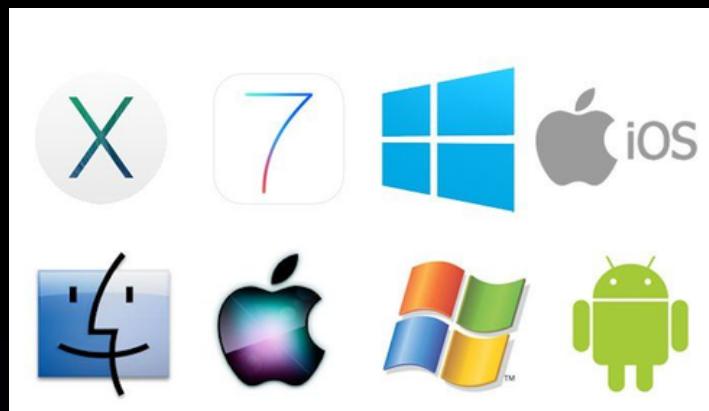
- No single processor
- should have separate buses.

**Question : What changes would be required to this program code in order to transfer it into Massively Parallel Computer?**

- Split into blocks of code so that it can be processed simultaneously instead of sequentially.
- Each block is processed by a different processor which allows each of the processor to simultaneously process the different blocks of code independently.



# **Virtual Machine**



**Question : What is a virtual machine manager?**

- **It is a type of software that allows us to run an operating system within another operating system.**

**Question : Explain the difference between Guest operating system and Host Operating System ?**

**Host Operating System:**

**The operating system that is actually controlling the physical hardware, or the operating system that is running virtual machine software.**

**Guest Operating System:**

**An operating system running in a virtual machine and controls virtual hardware.**

**Guest OS is running under the host OS software.**

**Question : What is meant by a virtual machine?**

- **The emulation of a computer system, including hardware and software.**
- **Uses a host computer system to run the virtual environment.**
- **Runs guest operating system(s) for emulation.**

## **Question : What are the benefits and drawbacks of virtual machine?**

### **Benefits**

- **Multiple guest operating systems can be used on the same computer.**
- **Different instruction set architectures can be emulated on a single computer.**
- **A virtual machine can crash without affecting the host machine.**
- **There are security benefits like trying a piece of suspicious software and if it has a virus, it will only infect the virtual machine.**
- **Cost savings due to not needing to purchase extra hardware.**
- **Can run Outdated applications that are currently incompatible and might have security risk**

### **Drawbacks**

- **A virtual machine has poorer performance than real machines because of extra load on the host computer.**
- **Performance of the guest system cannot be accurately measured.**
- **A virtual machine may be affected by any weaknesses of the host machine.**
- **Costly and complex to maintain**
- **Cannot emulate some hardware.**

# App Testing Through Virtual Machine

**Question : Describe the role of Virtual Machine Software in testing of an App?**

- By virtual machine software, you can create and manage virtual machines.
- Translate instructions used by guest operating system to that required by host OS.
- Emulates hardware.
- Protect each virtual machine so instances of the app can be tested together.

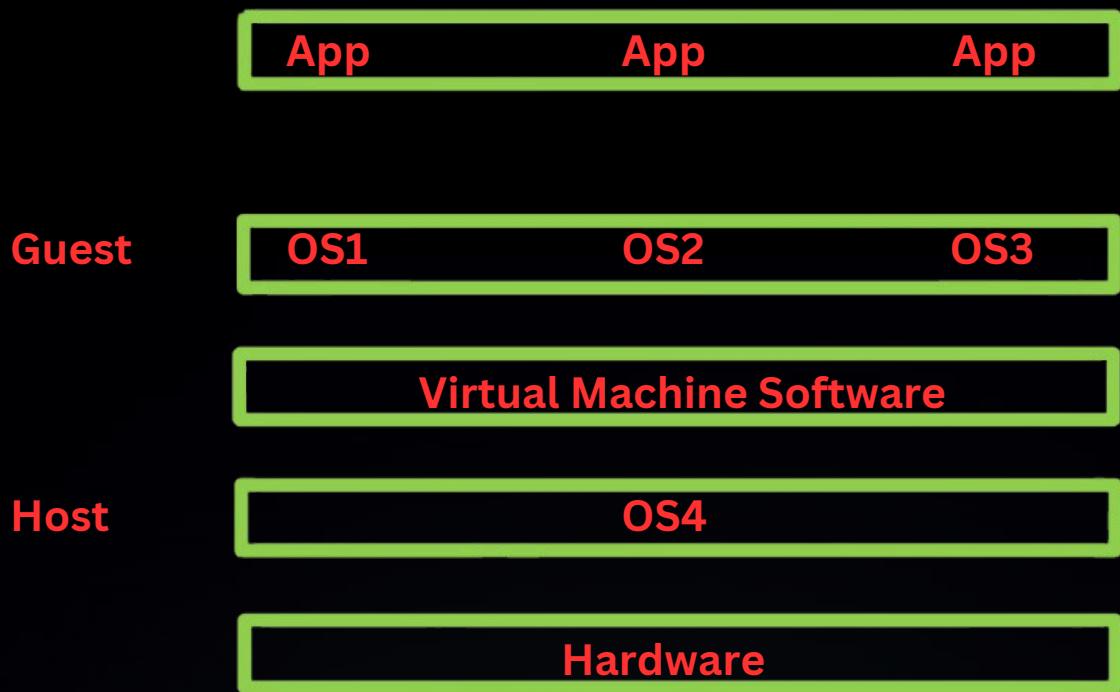
**Question : Benefits and drawbacks of testing by this approach ?**

**Benefits:**

- Multiple OS can exist simultaneously, allowing for testing using the same hardware.
- Only one set of hardware is required.
- Reduces the cost of producing the app as there will be no need to set up more than one computer.

**Drawbacks:**

- Execution of extra code.
- So performance is degraded and more time is taken to create the app and cannot make judgments about response time.



**Question : Describe the role of the four Operating Systems?**

- OS1, OS2, OS3 are guest operating systems secondary to one installed on the hardware.
- OS4 is the host operating system which interacts directly with the computer hardware.
- My App needs to run on all three guest OS with identical results.

**Note:** To identify host OS, look for OS which is in direct contact with hardware.

**Note:** A virtual machine is a software that emulates a different computer system. A virtual machine allows multiple Guest operating systems to run on one computer using a host operating system.

# **Software Production Through Virtual Machine**

**Question : State benefits of using VM to produce software ?**

- Software can be tried on different OS using the same hardware.
- No need to purchase different types of hardware.
- Easier to recover if software causes system crash.
- Virtual machine provides protection to other software.

**Question : Limitations in producing software by VM?**

- Using VM means execution of extra code, so processing time is increased due to which speed of real performance cannot be tested accordingly.
- A virtual machine might not be as efficient because it might not be able to access sufficient memory.
- VM may not be able to emulate the same hardware, due to which hardware cannot be tested using a virtual machine.

**Question : What happens if guest OS receives the data request from an application?**

- Guest OS handles the request as if it were running on its own physical machine and Guest OS is not aware that it is running on a virtual machine.
- Guest OS handles the request as usual.
- I/O requests are translated by virtual machine software into instructions executed by host OS.
- Host OS retrieves the data from the file.
- Host OS passes the data to virtual machine software.
- The virtual machine passes the data to Guest OS.
- Guest OS passes the data to the application.

**Question : What are the tasks performed by VM software?**

- Creates VM.
- Hardware emulation (Existing hardware is made available to guest OS.)
- Controls virtual hardware.
- Ensures each virtual machine is protected from actions of another VM.

## **Webserver Through Virtual Machine**

**Question : Describe uses of VM by Webserver company??**

- Could use alternative OS to identify possible problems, and it is much easier to create a VM with a new operating system rather than creating a new computer system.
- Could provide a safe environment during testing, which does not disrupt the web server services.
- Could use alternative replacement web-server software to identify possible problems in the webserver on different OS. It is easier to try different webserver and OS combinations.

# **Hardware and virtual machines**

## **Question 1**

- 5** Complete these **three** statements about computer processors.

A processor with a few simple fixed-length instructions that have a small number of instruction formats is called a ..... processor.

A processor with many complex variable-length instructions that has many instruction formats is called a ..... processor.

Instruction-level parallelism, applied to the execution of instructions during the fetch-execute cycle, is called .....

[3]

## **Question 2**

- 5** Complete these statements about a virtual machine.

A virtual machine is ..... that emulates a ..... computer system.

A virtual machine allows multiple ..... operating systems to run on one computer using a ..... operating system.

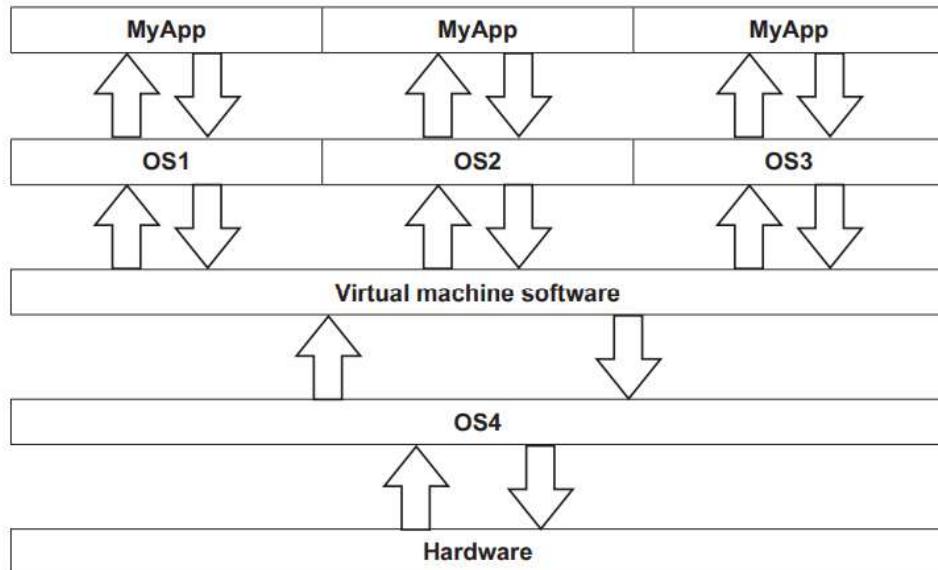
[4]

### Question 3

- 6 Mahmoud is developing a new application, MyApp, that needs to work with three different operating systems (OS1, OS2 and OS3).

He has decided to use virtual machine software to test MyApp with these three different operating systems.

The diagram shows MyApp running on three virtual machines.



- (a) Describe the roles of the **four** operating systems (OS1, OS2, OS3 and OS4) shown in the diagram.

.....  
.....  
.....  
.....  
.....

[3]

- (b) Describe the role of the virtual machine software in the testing of MyApp.

.....  
.....  
.....  
.....  
.....

[3]

- (c) Explain **one** benefit and **one** drawback of this approach to testing MyApp.

Benefit .....

---

---

---

Drawback .....

---

---

---

[4]

## Question 4

- 7 (a) RISC (Reduced Instruction Set Computing) and CISC (Complex Instruction Set Computing) are two types of processor.

Tick () **one** box in each row to show if the statement applies to RISC or CISC processors.

Statement	RISC	CISC
Larger instruction set		
Variable length instructions		
Smaller number of instruction formats		
Pipelining is easier		
Microprogrammed control unit		
Multi-cycle instructions		

[3]

- (b) In parallel processing, a computer can have multiple processors running in parallel.

- (i) State the **four** basic computer architectures used in parallel processing.

1 .....

2 .....

3 .....

4 .....

[4]

- (ii) Describe what is meant by a **massively parallel computer**.

.....  
.....  
.....  
.....  
.....

[3]

## Question 5

- 9 (a) The following incomplete table shows descriptions relating to computer architectures.

Complete the table by inserting the appropriate terms.

	Description	Term
A	<ul style="list-style-type: none"><li>• There are several processors.</li><li>• Each processor executes different sets of instructions on one set of data at the same time.</li></ul>	.....
B	<ul style="list-style-type: none"><li>• The processor has several ALUs.</li><li>• Each ALU executes the same set of instructions on different sets of data at the same time.</li></ul>	.....
C	<ul style="list-style-type: none"><li>• There is only one processor.</li><li>• The processor executes one set of instructions on one set of data.</li></ul>	.....
D	<ul style="list-style-type: none"><li>• There are several processors.</li><li>• Each processor executes a different set of instructions.</li><li>• Each processor operates on different sets of data.</li></ul>	.....

[4]

- (b) State **three** characteristics of massively parallel computers.

1 .....

.....

2 .....

.....

3 .....

[3]

## Question 6

- 5 (a) Most desktop or laptop computers use CISC (Complex Instruction Set Computing) architecture. Most smartphones and tablets use RISC (Reduced Instruction Set Computing).

State **four** features that are different for the CISC and RISC architectures.

1 .....

2 .....

3 .....

4 .....

[4]

- (b) In a RISC processor, four instructions (**A, B, C, D**) are processed using pipelining.

The following table shows five stages that take place when instructions are fetched and executed. In time interval 1, instruction **A** has been fetched.

- (i) In the table, write the instruction labels (**A, B, C, D**) in the correct time interval for each stage. Each operation only takes one time interval.

Stage	Time interval								
	1	2	3	4	5	6	7	8	9
Fetch instruction	<b>A</b>								
Decode instruction									
Execute instruction									
Access operand in memory									
Write result to register									

[3]

- (ii) When completed, the table in **part (b)(i)** shows how pipelining allows instructions to be carried out more rapidly. Each time interval represents one clock cycle.

Calculate how many clock cycles are saved by using pipelining in the example in **part (b)(i)**.

Show your working.

Working .....

.....

.....

Answer .....

[3]

- (c) The table shows four statements about computer architecture.

Put a tick ( $\checkmark$ ) in each row to identify the computer architecture associated with each statement.

Statement	Architecture		
	SIMD	MIMD	SISD
Each processor executes a different instruction			
There is only one processor			
Each processor executes the same instruction input using data available in the dedicated memory			
Each processor typically has its own partition within a shared memory			

[4]

## Question 7

- 2 (a) The following diagram shows four descriptions and four types of computer architecture.

Draw lines to connect each description to the appropriate computer architecture.

Description	Computer architecture
Most parallel computer systems use this architecture.	SIMD
Widely used to process 3D graphics in video games.	MIMD
A microprocessor is used to control a washing machine.	MISD
There are a number of processing units. Each processing unit executes the same instruction but on different data.	SISD

[4]

- (b) A computer has a single processor that contains four processing units.

Explain why this is **not** an example of a massively parallel computer.

.....

.....

.....

.....

[2]

- (c) An application has previously executed on a single computer. The application will be transferred onto a massively parallel computer.

The program code used in the application will need to be updated to ensure that the power of the massively parallel computer is fully used.

Explain what changes will be required to the program code.

.....  
.....  
.....

[2]

- (d) Explain **one** of the hardware issues that will have to be overcome if a massively parallel computer is to function successfully.

.....  
.....  
.....

[2]

## Question 8

- 2 (a) The following diagram shows four descriptions and two types of processor.

Draw lines to connect each description to the appropriate type of processor.

Description	Type of processor
It has a simplified set of instructions.	
Emphasis is on the hardware rather than the software.	CISC
It makes extensive use of general purpose registers.	RISC
Many instruction formats are available.	

[4]

- (b) In a RISC processor, instructions are processed using pipelining.

- (i) Explain what is meant by **pipelining**.

.....  
.....  
.....  
.....

[2]

- (ii) The following table shows the five stages that occur when instructions are fetched and executed. The table also shows a number of time intervals.

Two instructions, D followed by E, are fetched and executed. The 'E' in the incomplete table shows that instruction E has been fetched in time interval 2.

Complete each row of the table.

Stage	Time interval							
	1	2	3	4	5	6	7	8
Fetch instruction		E						
Read registers and decode instruction								
Execute instruction								
Access operand in memory								
Write result to register								

[3]

- (c) The instruction set for a RISC processor that allows pipelining includes the following instruction.

Instruction		Explanation
Op code	Operands	
ADD	<dest>, <op1>, <op2>	Add the integers in registers op1 and op2. Place the result in register dest.

A program contains the following three instructions.

ADD r3, r2, r1

ADD r5, r4, r3

ADD r10, r9, r8

- (i) Explain why pipelining fails for the first two instructions.

.....  
.....  
.....

[2]

- (ii) The instructions were produced by a compiler after translation of a high-level language program.

The compiler is not capable of code optimisation.

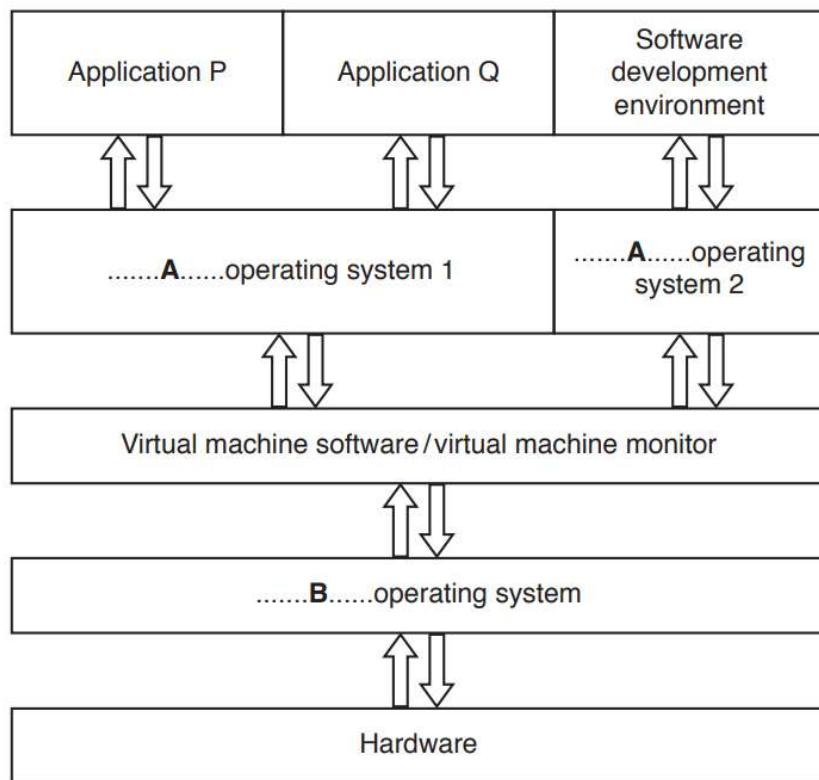
State how the code from the compiler could have been optimised to overcome the problem in part (c)(i).

.....  
.....

[1]

## Question 9

- 3 (a) This diagram shows how applications P, Q and a software development environment can be run on a virtual machine system.



- (i) State the operating systems labelled **A** and **B** in the diagram.

**A** .....

**B** .....

[2]

- (ii) Application P is executing and requests data from a file.

Describe what happens after .....**A**.....operating system 1 has received the data request from the application.

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

[3]

- (b) A software development company uses virtual machines to produce software.

- (i) State **one** benefit to the company.

.....  
.....

[1]

- (ii) Explain **two** limitations of this approach.

Limitation 1 .....

.....  
.....  
.....

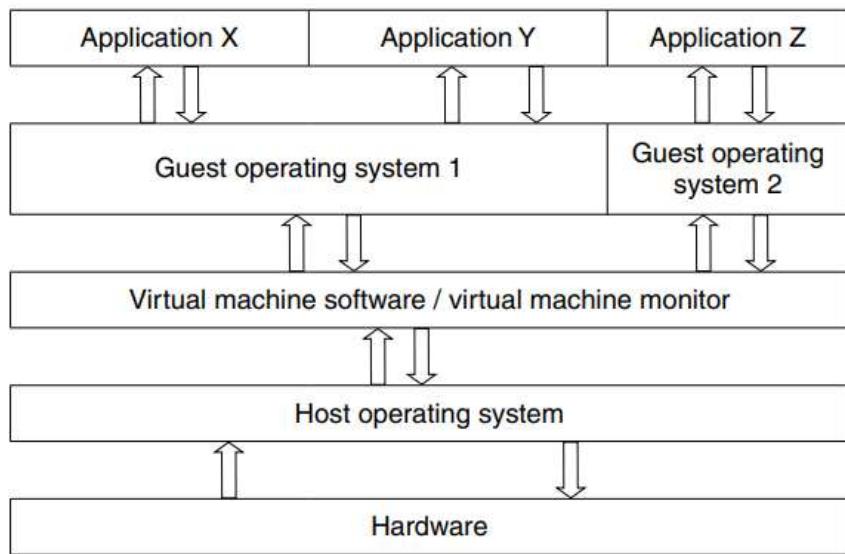
Limitation 2 .....

.....  
.....  
.....

[4]

## Question 10

- 3 (a) The following diagram shows how applications X, Y and Z can run on a virtual machine system.



- (i) The virtual machine software undertakes many tasks.

Describe **two** of these tasks.

Task 1 .....

.....

Task 2 .....

..... [2]

- (ii) Explain the difference between a **guest operating system** and a **host operating system**.

.....  
.....  
.....  
..... [2]

- (b) A company uses a computer as a web server. The manufacturer will no longer support the computer's operating system (OS) in six months' time. The company will then need to decide on a replacement OS.

The company is also considering changing the web server software when the OS is changed.

Whenever any changes are made, it is important that the web server service is not disrupted.

In developing these changes, the company could use virtual machines.

- (i) Describe **two** possible uses of virtual machines by the company.

Use 1 .....

.....  
.....  
.....

Use 2 .....

.....  
.....  
.....

[4]

The web server often has to handle many simultaneous requests.

- (ii) The company uses a virtual machine to test possible solutions to the changes that they will need to make.

Explain **one** limitation of this approach.

.....  
.....  
.....

[2]

## Question 11

- 4 (a) Three descriptions and two types of processor are shown below.

Draw a line to connect each description to the appropriate type of processor.

Description	Type of processor
Makes extensive use of general purpose registers	RISC
Many addressing modes are available	CISC
Has a simplified set of instructions	

[3]

- (b) In a RISC processor three instructions (A followed by B, followed by C) are processed using pipelining.

The following table shows the five stages that occur when instructions are fetched and executed.

- (i) The 'A' in the table indicates that instruction A has been fetched in time interval 1.

Complete the table to show the time interval in which each stage of each instruction (A, B, C) is carried out.

Stage	Time interval								
	1	2	3	4	5	6	7	8	9
Fetch instruction	A								
Decode instruction									
Execute instruction									
Access operand in memory									
Write result to register									

[3]

- (ii) The completed table shows how pipelining allows instructions to be carried out more rapidly. Each time interval represents one clock cycle.

Calculate how many clock cycles are saved by the use of pipelining in the above example.

Show your working.

.....

.....

.....

.....

.....

.....

.....

.....

[3]

## Question 12

- 4 (a) Four descriptions and four types of computer architecture are shown below.

Draw a line to connect each description to the appropriate type of computer architecture.

Description	Computer architecture
A computer that does not have the ability for parallel processing.	SIMD
The processor has several ALUs. Each ALU executes the same instruction but on different data.	MISD
There are several processors. Each processor executes different instructions drawn from a common pool. Each processor operates on different data drawn from a common pool.	SISD
There is only one processor executing one set of instructions on a single set of data.	MIMD

[4]

- (b) In a massively parallel computer explain what is meant by:

(i) Massive .....

.....

.....

[1]

(ii) Parallel .....

.....

.....

[1]

- (c) There are both hardware and software issues that have to be considered for parallel processing to succeed.

Describe **one** hardware and **one** software issue.

Hardware .....

.....

.....

Software .....

.....

.....

[4]

## Question 13

- 4 Reduced Instruction Set Computers (RISC) and Complex Instruction Set Computers (CISC) are two types of processor.

- (a) Describe what is meant by **RISC** and **CISC processors**.

RISC .....

.....

.....

CISC .....

.....

.....

[4]

- (b) Identify **two** differences between RISC and CISC processors.

1 .....

.....

.....

2 .....

.....

.....

[2]

## Question 14

- 6 A virtual machine is used to emulate a new computer system.

Describe **two** benefits and **one** limitation of using a virtual machine for this purpose.

Benefit 1 .....

.....  
.....  
.....  
.....

Benefit 2 .....

.....  
.....  
.....  
.....

Limitation .....

.....  
.....  
.....  
.....

[6]

## Question 15

- 5 There are four basic categories of computer architecture. Single Instruction Single Data (SISD) is one architecture.

Identify the **three other** categories of computer architecture.

Describe each category that you identify.

Architecture 1 .....

Description .....

.....  
.....  
.....

Architecture 2 .....

Description .....

.....  
.....  
.....

Architecture 3 .....

Description .....

.....  
.....  
.....

[6]

## Question 16

- 10 Reduced Instruction Set Computers (RISC) and Complex Instruction Set Computers (CISC) are two types of processor.

- (a) Tick ( $\checkmark$ ) **one** box in each row to show if the statement applies to RISC or CISC processors.

Statement	RISC	CISC
uses a smaller instruction set		
uses single-cycle instructions and limited addressing modes		
uses fewer general-purpose registers		
uses both hardwired and micro-coded control unit		
uses a system where cache is split between data and instructions		

[2]

- (b) Describe the process of pipelining during the fetch-execute cycle in RISC processors.

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

[4]

## Question 17

- 5 Describe what is meant by a **virtual machine**.

Include in your answer **two** benefits and **two** drawbacks of using a virtual machine.

Description .....

.....  
.....  
.....

Benefit 1 .....

.....  
.....

Benefit 2 .....

.....  
.....

Drawback 1 .....

.....  
.....

Drawback 2 .....

[6]

## Question 18

- 8 Outline the characteristics of massively parallel computers.

.....  
.....  
.....  
.....  
.....  
.....  
.....

[3]

## Question 19

- 8 (a) Describe the use of pipelining in Reduced Instruction Set Computers (RISC).

.....  
.....  
.....  
.....

[2]

- (b) The processing of instructions is divided into five stages:

- instruction fetch (IF)
- instruction decode (ID)
- operand fetch (OF)
- instruction execute (IE)
- write back result (WB)

Each stage is carried out using a different register when pipelining is used.

Complete the table to show how a program consisting of **six** instructions would be completed using pipelining.

Processor stages	Clock cycles											
	1	2	3	4	5	6	7	8	9	10	11	12
	IF											
	ID											
	OF											
	IE											
	WB											

[4]

## Question 20

- 5 Describe the features of SISD and MIMD computer architectures.

SISD .....

.....  
.....  
.....

MIMD .....

.....  
.....  
.....

[4]

## Question 21

- 10 Describe the features of the SIMD and MISD computer architectures.

SIMD .....

.....  
.....  
.....

MISD .....

.....  
.....  
.....

[4]

## Question 22

- 11 Reduced Instruction Set Computers (RISC) and Complex Instruction Set Computers (CISC) are two types of processor.

- (a) State **two** features of RISC processors.

.....  
.....  
.....  
.....  
.....

[2]

- (b) Outline the process of interrupt handling as it could be applied to RISC or CISC processors.

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

[3]

- (c) Explain how pipelining affects interrupt handling for RISC processors.

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

[3]

## Question 23

- 9 (a) Outline **two** benefits and **two** limitations of a virtual machine.

Benefit 1 .....

.....  
.....

Benefit 2 .....

.....  
.....

Limitation 1 .....

.....  
.....

Limitation 2 .....

.....  
.....

[4]

- (b) Explain the roles of the host operating system and the guest operating system as used in a computer system running a virtual machine.

.....

.....

.....

..... [3]

# Answers

## Answer 1

5	<ul style="list-style-type: none"><li>• RISC / reduced instruction set computer</li><li>• CISC / complex instruction set computer</li><li>• Pipelining</li></ul>	3
---	--	---

## Answer 2

5	Software / a program Physical / different Guest Host	4
---	---	---

## Answer 3

6(a)	<p>Max <b>three</b>, <b>one</b> mark for role, <b>one</b> mark for expansion</p> <p>OS1, OS2 and OS3 are guest operating systems ... secondary to the one installed on the hardware</p> <p>OS4 is the host operating system ... interacts directly with the machine hardware</p> <p>MyApp needs to run on all three guest operating systems ... ... with identical results</p>	3
6(b)	<p>Any <b>three</b> from</p> <p>Create/delete/manage virtual machine</p> <p>Translate instructions used by guest operating system to that required by host operating system</p> <p>Hardware emulation</p> <p>Protecting each virtual machine ... so instances of MyApp can be tested together</p>	3
6(c)	<p><b>One</b> mark for benefit and <b>one</b> mark for relevant explanation</p> <p><b>One</b> mark for drawback <b>one</b> mark for relevant explanation</p> <p>For example:</p> <p>Benefit: multiple operating systems can exist simultaneously ... allowing for testing using the same hardware only one set of hardware required ... reduces cost of producing the app // no need to set up more than one computer</p> <p>Drawback: execution of extra code ... so performance is degraded // more time taken to execute the app // cannot make judgements about response time etc</p>	4

## Answer 4

7(a)	<p>1 mark for 2/3 rows correct      2 marks for 4/5 rows correct      3 marks for 6 correct rows</p> <table border="1"> <thead> <tr> <th>Statement</th><th>RISC</th><th>CISC</th></tr> </thead> <tbody> <tr> <td>Larger instruction set</td><td></td><td>✓</td></tr> <tr> <td>Variable length instructions</td><td></td><td>✓</td></tr> <tr> <td>Smaller number of instruction formats</td><td>✓</td><td></td></tr> <tr> <td>Pipelining is easier</td><td>✓</td><td></td></tr> <tr> <td>Microprogrammed control unit</td><td></td><td>✓</td></tr> <tr> <td>Multi-cycle instructions</td><td></td><td>✓</td></tr> </tbody> </table>	Statement	RISC	CISC	Larger instruction set		✓	Variable length instructions		✓	Smaller number of instruction formats	✓		Pipelining is easier	✓		Microprogrammed control unit		✓	Multi-cycle instructions		✓	3
Statement	RISC	CISC																					
Larger instruction set		✓																					
Variable length instructions		✓																					
Smaller number of instruction formats	✓																						
Pipelining is easier	✓																						
Microprogrammed control unit		✓																					
Multi-cycle instructions		✓																					
7(b)(i)	<p>1 mark per bullet point</p> <ul style="list-style-type: none"> <li>∞ SISD // Single instruction single data</li> <li>∞ SIMD // Single instruction multiple data</li> <li>∞ MISD // Multiple instruction single data</li> <li>∞ MIMD // Multiple instruction multiple data</li> </ul>	4																					
7(b)(ii)	<p>1 mark per bullet point (max 3)</p> <ul style="list-style-type: none"> <li>∞ <b>Large number</b> of processors</li> <li>∞ ... working collaboratively on the same program</li> <li>∞ ... working together simultaneously on the same program</li> <li>∞ ... communicating via a messaging interface</li> </ul>	3																					

## Answer 5

9(a)	<p><b>1 mark</b> for each correct term</p> <table border="1"> <thead> <tr> <th>Description</th><th>Term</th></tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>• There are several processors.</li> <li>• Each processor executes different sets of instructions on one set of data at the same time.</li> </ul> </td><td>MISD</td></tr> <tr> <td> <ul style="list-style-type: none"> <li>• The processor has several ALUs.</li> <li>• Each ALU executes the same set of instructions on different sets of data at the same time.</li> </ul> </td><td>SIMD</td></tr> <tr> <td> <ul style="list-style-type: none"> <li>• There is only one processor.</li> <li>• The processor executes one set of instructions on one set of data.</li> </ul> </td><td>SISD</td></tr> <tr> <td> <ul style="list-style-type: none"> <li>• There are several processors.</li> <li>• Each processor executes a different set of instructions.</li> <li>• Each processor operates on different sets of data.</li> </ul> </td><td>MIMD</td></tr> </tbody> </table>	Description	Term	<ul style="list-style-type: none"> <li>• There are several processors.</li> <li>• Each processor executes different sets of instructions on one set of data at the same time.</li> </ul>	MISD	<ul style="list-style-type: none"> <li>• The processor has several ALUs.</li> <li>• Each ALU executes the same set of instructions on different sets of data at the same time.</li> </ul>	SIMD	<ul style="list-style-type: none"> <li>• There is only one processor.</li> <li>• The processor executes one set of instructions on one set of data.</li> </ul>	SISD	<ul style="list-style-type: none"> <li>• There are several processors.</li> <li>• Each processor executes a different set of instructions.</li> <li>• Each processor operates on different sets of data.</li> </ul>	MIMD	4
Description	Term											
<ul style="list-style-type: none"> <li>• There are several processors.</li> <li>• Each processor executes different sets of instructions on one set of data at the same time.</li> </ul>	MISD											
<ul style="list-style-type: none"> <li>• The processor has several ALUs.</li> <li>• Each ALU executes the same set of instructions on different sets of data at the same time.</li> </ul>	SIMD											
<ul style="list-style-type: none"> <li>• There is only one processor.</li> <li>• The processor executes one set of instructions on one set of data.</li> </ul>	SISD											
<ul style="list-style-type: none"> <li>• There are several processors.</li> <li>• Each processor executes a different set of instructions.</li> <li>• Each processor operates on different sets of data.</li> </ul>	MIMD											

9(b)	<p><b>1 mark per bullet point to max 3</b></p> <ul style="list-style-type: none"> <li>• A <b>large number</b> of processors</li> <li>• Collaborative processing // coordinated simultaneous processing</li> <li>• Network infrastructure</li> <li>• Communicate using a message interface / by sending messages</li> </ul>	<b>3</b>
------	--	----------

## Answer 6

5(a)	<p>1 mark per bullet point to max 4:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"> <ul style="list-style-type: none"> <li>• RISC has fewer instructions</li> <li>• RISC has many registers</li> <li>• RISC's instructions are simpler</li> <li>• RISC has a few instruction formats</li> <li>• RISC usually uses single-cycle instructions</li> <li>• RISC uses fixed-length instructions</li> <li>• RISC has better pipelineability</li> <li>• RISC requires less complex circuits</li> <li>• RISC has fewer addressing modes</li> <li>• RISC makes more use of RAM</li> <li>• RISC has a hard-wired control unit</li> <li>• RISC only uses load and store instructions to address memory</li> </ul> </td><td style="width: 50%;"> <ul style="list-style-type: none"> <li>// CISC has more instructions</li> <li>// CISC has few registers</li> <li>// CISC's instructions are more complex</li> <li>// CISC has many instruction formats</li> <li>// CISC uses multi-cycle instructions</li> <li>// CISC uses variable-length instructions</li> <li>// CISC has poorer pipelineability</li> <li>// CISC requires more complex circuits</li> <li>// CISC has more addressing modes</li> <li>// CISC makes more use of cache/less use of RAM</li> <li>// CISC has a programmable control unit</li> </ul> </td></tr> </table>	<ul style="list-style-type: none"> <li>• RISC has fewer instructions</li> <li>• RISC has many registers</li> <li>• RISC's instructions are simpler</li> <li>• RISC has a few instruction formats</li> <li>• RISC usually uses single-cycle instructions</li> <li>• RISC uses fixed-length instructions</li> <li>• RISC has better pipelineability</li> <li>• RISC requires less complex circuits</li> <li>• RISC has fewer addressing modes</li> <li>• RISC makes more use of RAM</li> <li>• RISC has a hard-wired control unit</li> <li>• RISC only uses load and store instructions to address memory</li> </ul>	<ul style="list-style-type: none"> <li>// CISC has more instructions</li> <li>// CISC has few registers</li> <li>// CISC's instructions are more complex</li> <li>// CISC has many instruction formats</li> <li>// CISC uses multi-cycle instructions</li> <li>// CISC uses variable-length instructions</li> <li>// CISC has poorer pipelineability</li> <li>// CISC requires more complex circuits</li> <li>// CISC has more addressing modes</li> <li>// CISC makes more use of cache/less use of RAM</li> <li>// CISC has a programmable control unit</li> </ul>
<ul style="list-style-type: none"> <li>• RISC has fewer instructions</li> <li>• RISC has many registers</li> <li>• RISC's instructions are simpler</li> <li>• RISC has a few instruction formats</li> <li>• RISC usually uses single-cycle instructions</li> <li>• RISC uses fixed-length instructions</li> <li>• RISC has better pipelineability</li> <li>• RISC requires less complex circuits</li> <li>• RISC has fewer addressing modes</li> <li>• RISC makes more use of RAM</li> <li>• RISC has a hard-wired control unit</li> <li>• RISC only uses load and store instructions to address memory</li> </ul>	<ul style="list-style-type: none"> <li>// CISC has more instructions</li> <li>// CISC has few registers</li> <li>// CISC's instructions are more complex</li> <li>// CISC has many instruction formats</li> <li>// CISC uses multi-cycle instructions</li> <li>// CISC uses variable-length instructions</li> <li>// CISC has poorer pipelineability</li> <li>// CISC requires more complex circuits</li> <li>// CISC has more addressing modes</li> <li>// CISC makes more use of cache/less use of RAM</li> <li>// CISC has a programmable control unit</li> </ul>		

5(b)(i)	<p>1 mark per bullet point:</p> <ul style="list-style-type: none"> <li>• Completing the As correctly</li> <li>• B in column 2, row 1 no other Bs in row 1</li> <li>• Remainder correctly completed</li> </ul>																																																																				
	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th rowspan="2" style="text-align: center; width: 15%;">Stage</th> <th colspan="9" style="text-align: center;">Time interval</th> </tr> <tr> <th style="text-align: center;">1</th> <th style="text-align: center;">2</th> <th style="text-align: center;">3</th> <th style="text-align: center;">4</th> <th style="text-align: center;">5</th> <th style="text-align: center;">6</th> <th style="text-align: center;">7</th> <th style="text-align: center;">8</th> <th style="text-align: center;">9</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;"><b>Fetch instruction</b></td><td style="text-align: center;"><b>A</b></td><td style="text-align: center;"><b>B</b></td><td style="text-align: center;"><b>C</b></td><td style="text-align: center;"><b>D</b></td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td style="text-align: center;"><b>Decode instruction</b></td><td></td><td style="text-align: center;"><b>A</b></td><td style="text-align: center;"><b>B</b></td><td style="text-align: center;"><b>C</b></td><td style="text-align: center;"><b>D</b></td><td></td><td></td><td></td><td></td></tr> <tr> <td style="text-align: center;"><b>Execute instruction</b></td><td></td><td></td><td style="text-align: center;"><b>A</b></td><td style="text-align: center;"><b>B</b></td><td style="text-align: center;"><b>C</b></td><td style="text-align: center;"><b>D</b></td><td></td><td></td><td></td></tr> <tr> <td style="text-align: center;"><b>Access operand in memory</b></td><td></td><td></td><td></td><td style="text-align: center;"><b>A</b></td><td style="text-align: center;"><b>B</b></td><td style="text-align: center;"><b>C</b></td><td style="text-align: center;"><b>D</b></td><td></td><td></td></tr> <tr> <td style="text-align: center;"><b>Write result to register</b></td><td></td><td></td><td></td><td></td><td style="text-align: center;"><b>A</b></td><td style="text-align: center;"><b>B</b></td><td style="text-align: center;"><b>C</b></td><td style="text-align: center;"><b>D</b></td><td></td></tr> </tbody> </table>	Stage	Time interval									1	2	3	4	5	6	7	8	9	<b>Fetch instruction</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>						<b>Decode instruction</b>		<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>					<b>Execute instruction</b>			<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>				<b>Access operand in memory</b>				<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>			<b>Write result to register</b>					<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
Stage	Time interval																																																																				
	1	2	3	4	5	6	7	8	9																																																												
<b>Fetch instruction</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>																																																																	
<b>Decode instruction</b>		<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>																																																																
<b>Execute instruction</b>			<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>																																																															
<b>Access operand in memory</b>				<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>																																																														
<b>Write result to register</b>					<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>																																																													

5(b)(ii)	<p>1 mark per bullet point:</p> <ul style="list-style-type: none"> <li>• Correct number of cycles for pipelining 8</li> <li>• Correct number of cycles without pipelining <math>4 \times 5 = 20</math></li> <li>• No of cycles saved <math>20 - 8 = 12</math></li> </ul>
----------	--

5(c)	1 mark for each row																							
	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th rowspan="2" style="padding: 2px;">Statement</th> <th colspan="3" style="padding: 2px;">Architecture</th> </tr> <tr> <th style="padding: 2px;">SIMD</th> <th style="padding: 2px;">MIMD</th> <th style="padding: 2px;">SISD</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">Each processor executes a different instruction</td><td style="padding: 2px;"></td><td style="padding: 2px;">✓</td><td style="padding: 2px;"></td></tr> <tr> <td style="padding: 2px;">There is only one processor</td><td style="padding: 2px;"></td><td style="padding: 2px;"></td><td style="padding: 2px;">✓</td></tr> <tr> <td style="padding: 2px;">Each processor executes the same instruction input using data available in the dedicated memory</td><td style="padding: 2px;">✓</td><td style="padding: 2px;"></td><td style="padding: 2px;"></td></tr> <tr> <td style="padding: 2px;">Each processor typically has its own partition within a shared memory</td><td style="padding: 2px;"></td><td style="padding: 2px;">✓</td><td style="padding: 2px;"></td></tr> </tbody> </table>	Statement	Architecture			SIMD	MIMD	SISD	Each processor executes a different instruction		✓		There is only one processor			✓	Each processor executes the same instruction input using data available in the dedicated memory	✓			Each processor typically has its own partition within a shared memory		✓	
Statement	Architecture																							
	SIMD	MIMD	SISD																					
Each processor executes a different instruction		✓																						
There is only one processor			✓																					
Each processor executes the same instruction input using data available in the dedicated memory	✓																							
Each processor typically has its own partition within a shared memory		✓																						

## Answer 7

2(a)	<table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Description</th><th style="width: 50%;">Computer architecture</th></tr> </thead> <tbody> <tr> <td style="border: 1px solid black; padding: 5px;">Most parallel computer systems use this architecture.</td><td style="border: 1px solid black; padding: 5px; text-align: center;">SIMD</td></tr> <tr> <td style="border: 1px solid black; padding: 5px;">Widely used to process 3D graphics in video games.</td><td style="border: 1px solid black; padding: 5px; text-align: center;">MIMD</td></tr> <tr> <td style="border: 1px solid black; padding: 5px;">A microprocessor is used to control a washing machine.</td><td style="border: 1px solid black; padding: 5px; text-align: center;">MISD</td></tr> <tr> <td style="border: 1px solid black; padding: 5px;">There are a number of processing units. Each processing unit executes the same instruction but on different data</td><td style="border: 1px solid black; padding: 5px; text-align: center;">SISD</td></tr> </tbody> </table> <p style="text-align: right; margin-top: -10px;">1 mark for each correct line</p>	Description	Computer architecture	Most parallel computer systems use this architecture.	SIMD	Widely used to process 3D graphics in video games.	MIMD	A microprocessor is used to control a washing machine.	MISD	There are a number of processing units. Each processing unit executes the same instruction but on different data	SISD	4
Description	Computer architecture											
Most parallel computer systems use this architecture.	SIMD											
Widely used to process 3D graphics in video games.	MIMD											
A microprocessor is used to control a washing machine.	MISD											
There are a number of processing units. Each processing unit executes the same instruction but on different data	SISD											
2(b)	<ul style="list-style-type: none"> <li>∞ Only one (separate) processor / not many separate processors (is not massively parallel) <span style="float: right;">1</span></li> <li>∞ Quad core computer system // processing units share the same bus <span style="float: right;">1</span></li> </ul> <p style="text-align: right; margin-top: -10px;"><b>1 mark for each point, max 2</b></p>	2										
2(c)	<ul style="list-style-type: none"> <li>∞ Split into blocks of code ....</li> <li>∞ ... that can be processed simultaneously ...</li> <li>∞ ... instead of sequentially</li> <li>∞ Each block is processed by a different processor</li> <li>∞ which allows each of the many processors to simultaneously process the different blocks of code independently</li> <li>∞ Requires both parallelism and co-ordination</li> </ul> <p style="text-align: right; margin-top: -10px;"><b>1 mark for each point, max 2</b></p>	2										

2(d)	<p><b>1 mark</b> for identification of hardware issue, for example:</p> <ul style="list-style-type: none"> <li>∞ Communication between the different processors is the issue</li> </ul> <p><b>1 mark</b> for further explanation from:</p> <ul style="list-style-type: none"> <li>∞ Each processor needs a link to every other processor</li> <li>∞ Many processors require many of these links</li> <li>∞ Challenging topology</li> </ul>	2
------	--	---

## Answer 8

2(a)	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; width: 50%;">Description</th><th style="text-align: center; width: 50%;">Type of processor</th></tr> </thead> <tbody> <tr> <td style="border: 1px solid black; padding: 5px;">It has a simplified set of instructions.</td><td style="text-align: center; vertical-align: middle;"> </td></tr> <tr> <td style="border: 1px solid black; padding: 5px;">Emphasis is on the hardware rather than the software.</td><td style="text-align: center; vertical-align: middle;"></td></tr> <tr> <td style="border: 1px solid black; padding: 5px;">It makes extensive use of general purpose registers.</td><td style="text-align: center; vertical-align: middle;"></td></tr> <tr> <td style="border: 1px solid black; padding: 5px;">Many instruction formats are available.</td><td style="text-align: center; vertical-align: middle;"></td></tr> </tbody> </table> <p style="text-align: center;">1 mark for each correct line</p>	Description	Type of processor	It has a simplified set of instructions.		Emphasis is on the hardware rather than the software.		It makes extensive use of general purpose registers.		Many instruction formats are available.		4
Description	Type of processor											
It has a simplified set of instructions.												
Emphasis is on the hardware rather than the software.												
It makes extensive use of general purpose registers.												
Many instruction formats are available.												
2(b)(i)	<p><b>One mark per point – max 2</b></p> <ul style="list-style-type: none"> <li>∞ Pipelining is instruction level parallelism</li> <li>∞ Execution (A: processing) of an instruction is split into a number of stages</li> <li>∞ When first stage for an instruction is completed the first stage of the next instruction can start executing</li> <li>∞ Another instruction can start executing before the previous one is finished</li> <li>∞ Processing of a number of instructions can be concurrent / simultaneous</li> </ul>	2										

2(b)(ii)	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th rowspan="2" style="text-align: center;">Stage</th><th colspan="8" style="text-align: center;">Time Interval</th></tr> <tr> <th style="text-align: center;">1</th><th style="text-align: center;">2</th><th style="text-align: center;">3</th><th style="text-align: center;">4</th><th style="text-align: center;">5</th><th style="text-align: center;">6</th><th style="text-align: center;">7</th><th style="text-align: center;">8</th></tr> </thead> <tbody> <tr> <td>Fetch instruction</td><td style="text-align: center;">D</td><td style="text-align: center;"><b>E</b></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>Read registers and decode instruction</td><td></td><td style="text-align: center;">D</td><td style="text-align: center;">E</td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>Execute instruction</td><td></td><td></td><td style="text-align: center;">D</td><td style="text-align: center;">E</td><td></td><td></td><td></td><td></td></tr> <tr> <td>Access operand in memory</td><td></td><td></td><td></td><td style="text-align: center;">D</td><td style="text-align: center;">E</td><td></td><td></td><td></td></tr> <tr> <td>Write result to register</td><td></td><td></td><td></td><td></td><td style="text-align: center;">D</td><td style="text-align: center;">E</td><td></td><td></td></tr> </tbody> </table>	Stage	Time Interval								1	2	3	4	5	6	7	8	Fetch instruction	D	<b>E</b>							Read registers and decode instruction		D	E						Execute instruction			D	E					Access operand in memory				D	E				Write result to register					D	E			3
Stage	Time Interval																																																															
	1	2	3	4	5	6	7	8																																																								
Fetch instruction	D	<b>E</b>																																																														
Read registers and decode instruction		D	E																																																													
Execute instruction			D	E																																																												
Access operand in memory				D	E																																																											
Write result to register					D	E																																																										
<p>D at time interval 1 (1)            D and E in second row (in that order) (1)            Remainder completed correctly (1)</p>																																																																
2(c)(i)	<b>Two from:</b> <ul style="list-style-type: none"> <li>∞ The result of the first addition is not stored in (register) r3 (1)</li> <li>∞ Before the next instruction needs to load value from r3 (1)</li> <li>∞ There is a data dependency issue (1)</li> <li>∞ r3 is being fetched and stored on the same clock pulse (1)</li> </ul>	2																																																														
2(c)(ii)	The third instruction is not dependent on the first two, therefore, instruction 2 and 3 need to be swapped	1																																																														

## Abswer 9

3(a)(i)	<b>A:</b> Guest (operating system) (1) <b>B:</b> Host (operating system) (1)	2
3(a)(ii)	<b>One mark</b> for each valid point, <b>max 3</b> <ul style="list-style-type: none"> <li>∞ Guest OS (A) handles request as if it were running on its own physical machine // guest OS (A) is not aware it is running on a virtual platform</li> <li>∞ Guest OS (A) handles the request as usual</li> <li>∞ I/O requests are translated by the virtual machine software</li> <li>∞ Into instructions executed by host OS (B)</li> <li>∞ Host OS (<b>B</b>) retrieves the data from the file</li> <li>∞ Host OS (<b>B</b>) passes the data to the virtual machine software</li> <li>∞ The virtual machine software passes the data to the guest OS (<b>A</b>)</li> <li>∞ Guest OS passes the data to the application</li> </ul>	3
3(b)(i)	<b>One mark</b> from: <ul style="list-style-type: none"> <li>∞ Because software can be tried on different OS using same hardware</li> <li>∞ Because no need to purchase / request all sorts of different hardware</li> <li>∞ Easier to recover if software causes system crash</li> <li>∞ VM provides protection to other software / host OS from malfunctioning software</li> </ul>	1

3(b)(ii)	<p><b>Max 2 marks</b> per limitation, <b>max 2</b> limitations – <b>max 4 marks</b></p> <p>Virtual machine may not be able to emulate some hardware  ... So that hardware cannot be tested using a virtual machine  ... By relevant example, e.g. developing hardware drivers</p> <p>Using virtual machine means execution of extra code // processing time increased  ... so cannot accurately test speed of real performance</p> <p>A virtual machine might not be as efficient  ... By relevant example, e.g. might not be able to access sufficient memory</p>	4
----------	--	---

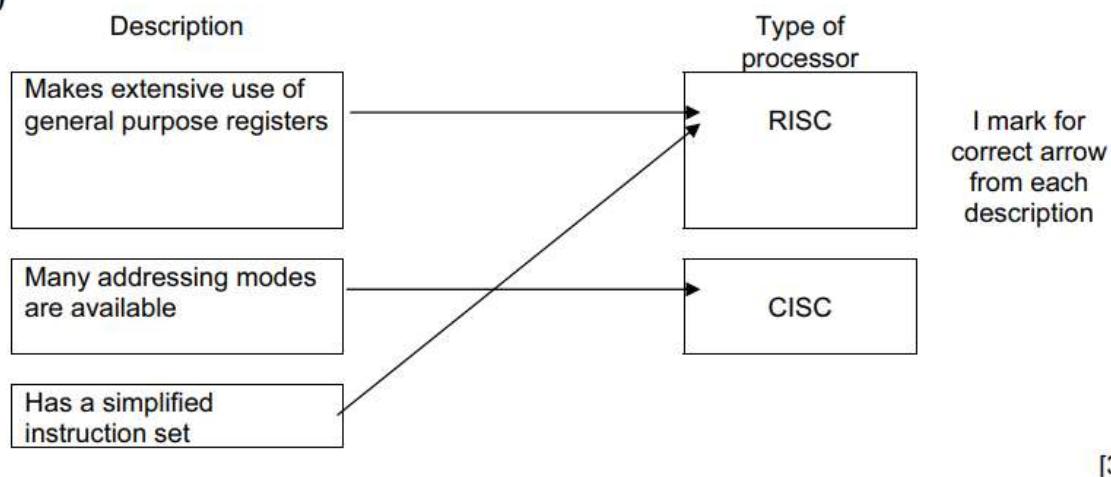
## Answer 10

3 (a) (i)	<p>Examples:  Create / delete virtual machine  Existing hardware made available to guest OS // hardware emulation  Ensures each virtual machine is protected from actions of another virtual machine</p>	1 1 1 <b>Max 2</b>
(ii)	<p>Guest operating system:  An operating system running in a virtual machine //  Controls virtual hardware //  OS is being emulated</p> <p>Host operating system:  The operating system that is actually controlling the physical hardware //  the operating system for the physical machine// the OS running the VM software</p> <p>Guest OS is running under the Host OS software</p>	1 1 1 <b>Max 2</b>
(b) (i)	<p>Examples:  Trial/use alternative replacement operating system(s) ...  Test to identify possible problems  Much easier to create VM with a new OS than create new computer system</p> <p>Trial/use alternative replacement web server software ...  Test to identify possible problems  Easier to try alternative new software <u>and</u> new OS combinations</p> <p>To provide some additional service(s)  Trial/test its use - description e.g. a print server</p> <p>General description point – to provide a safe environment during testing (which does not disrupt the web server service)</p>	<b>Two marks for each use</b> <b>Maximum two uses</b> <b>Max 4</b>

(ii)	<p>Examples: Using virtual machine means execution of extra code // emulation of some hardware ...</p> <p>Non-VM installation may not perform in the same way Execution speed slower than non-VM system Problems in judging actual response times at time of maximum traffic needs fastest possible speed</p> <p>Particular hardware may be difficult to emulate</p>	<p>1</p> <p>1</p> <p>1</p> <p>1</p> <p>1</p> <p><b>Max 2</b></p>
------	--	--

## Answer 11

4 (a)



(b) (i)

stage	Time Interval								
	1	2	3	4	5	6	7	8	9
Fetch instruction	A	B	C						
Decode instruction		A	B	C					
Execute instruction			A	B	C				
Access operand in memory				A	B	C			
Write result to register					A	B	C		

Completing the As (1 Mark)

B in column 2, Row 1 (1 Mark)

Remainder completed (1 Mark)

[3]

(ii) With pipelining no of cycles = 7

[1]

Without pipelining no of cycles =  $3 * 5 = 15$

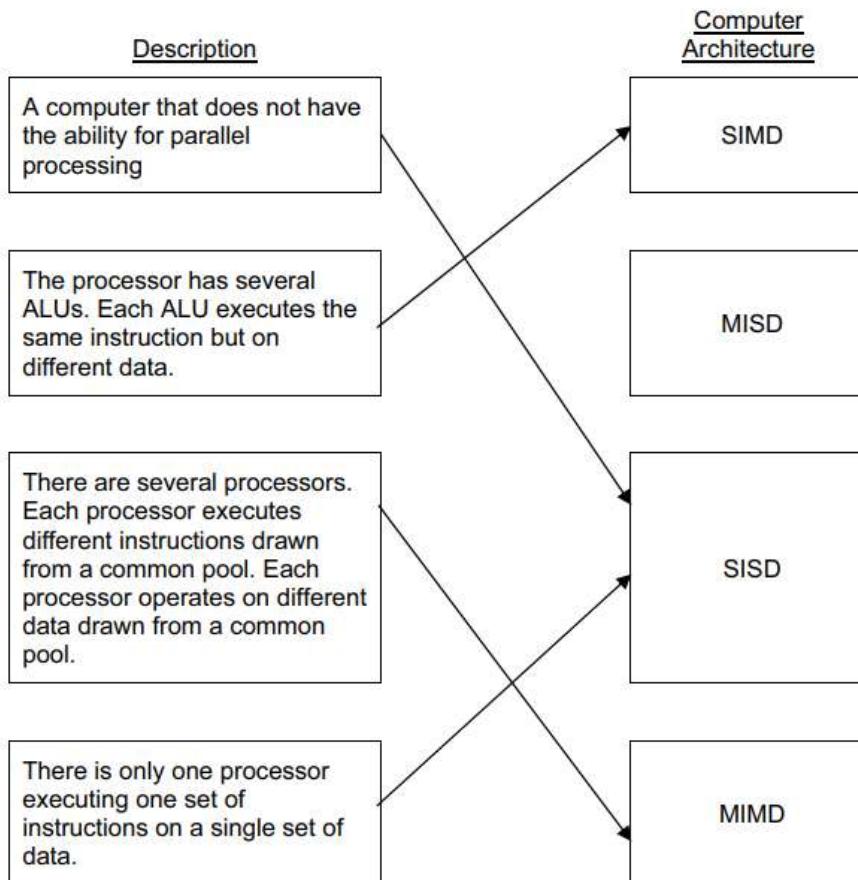
[1]

No of cycles saved = 8

[1]

## Answer 12

4 (a) 1 mark for correct arrow from each description



[4]

(b) (i) **Massive**: many/large number of processors // hundreds/thousands of processors [1]

(ii) **Parallel**: to perform a set of coordinated computations in parallel/simultaneously [1]

(c) processors need to be able to communicate ...  
so that processed data can be transferred from one processor to another [1]  
[1]

suitable algorithm/program/software/design // appropriate programming language  
which allows data to be processed by multiple processors simultaneously [1]  
[1]

## Answer 13

Question	Answer	Marks
4(a)	<p>RISC <b>max 2</b> any <b>two</b> from:</p> <ul style="list-style-type: none"> <li>• Uses simple instructions</li> <li>• Uses fixed length instructions</li> <li>• Instructions only require one clock cycle</li> <li>• Uses many registers</li> <li>• Makes use of pipelining</li> <li>• Hardwired CU</li> </ul> <p>CISC <b>max 2</b> any <b>two</b> from:</p> <ul style="list-style-type: none"> <li>• Uses many <b>instruction formats</b></li> <li>• Uses variable length instructions</li> <li>• Makes use of different addressing modes</li> <li>• Uses few registers</li> <li>• Has a large instruction set</li> <li>• Requires <b>complex circuits</b></li> <li>• Frequently uses cache</li> <li>• Instructions (converted to sub-instructions that) may require many clock cycles</li> <li>• Programmable CU</li> </ul>	4
4(b)	<p><b>One mark</b> for each difference <b>max 2</b> from:</p> <ul style="list-style-type: none"> <li>• RISC has fewer instructions // CISC has more instructions</li> <li>• RISC has many registers // CISC has few registers</li> <li>• RISC's instructions are simpler // CISC's instructions are more complex</li> <li>• RISC has a few instruction formats / CISC has many instruction formats</li> <li>• RISC usually uses single-cycle instructions// CISC uses multi-cycle instructions</li> <li>• RISC uses fixed-length instructions // CISC uses variable-length instructions</li> <li>• RISC has better pipelineability // CISC has poorer pipelineability</li> <li>• RISC requires less complex circuits// CISC requires more complex circuits</li> <li>• RISC has fewer addressing modes // CISC has more addressing modes</li> <li>• RISC makes more use of RAM// CISC makes more use of cache/less use of RAM</li> <li>• RISC has a hard-wired control unit // CISC has a programmable control unit</li> <li>• RISC only uses load and store instructions to address memory // CISC has many types of instructions to address memory</li> </ul>	2

## Answer 14

Question	Answer	Marks
6	<p><b>Two marks</b> each benefit description <b>max 4</b></p> <p>New system can be tried on different virtual hardware (1) without need to purchase the hardware (1)</p> <p>Easier to recover if software emulating the new computer causes system crash (1) as VM provides protection to other software (1)</p> <p>Emulate programs for the new computer system that are not compatible with the host computer / operating system (1) by using the guest operating system on the old computer (1)</p> <p>More than one new computer system can be emulated (1) this allows multiple operating systems to coexist on a single computer(1)</p> <p><b>Two marks</b> each limitation description <b>max 2</b> from:</p> <p>Virtual machines may not be able to emulate the new hardware (1) because this hardware may have been developed since the virtual machine was developed (1)</p> <p>Using virtual machine means execution of extra code // A virtual machine might not be as efficient // resources e.g. memory or processor time are shared (1) processing time increased // performance degrades (1)</p> <p>Use of a virtual machine increases the maintenance overheads (1) because both host system and the virtual machine must be maintained (1)</p>	6

## Answer 15

Question	Answer	Marks
5	SIMD (1) many/array processors execute the same instruction using different data sets (1) MISD (1) many processors (using different instructions) use the same data set (1) MIMD (1) many processors (using different instructions) using different data sets (1)	6

## Answer 16

Question	Answer																				
10(a)	<p>Two marks for all five rows correct One mark for four rows correct</p> <table border="1"> <thead> <tr> <th>Statement</th> <th>RISC</th> <th>CISC</th> </tr> </thead> <tbody> <tr> <td>uses a smaller instruction set</td> <td>✓</td> <td></td> </tr> <tr> <td>uses single-cycle instructions and limited addressing modes</td> <td>✓</td> <td></td> </tr> <tr> <td>uses fewer general-purpose registers</td> <td></td> <td>✓</td> </tr> <tr> <td>uses both hardwired and micro coded control unit</td> <td></td> <td>✓</td> </tr> <tr> <td>uses a system where cache is split between data and instructions</td> <td>✓</td> <td></td> </tr> </tbody> </table>	Statement	RISC	CISC	uses a smaller instruction set	✓		uses single-cycle instructions and limited addressing modes	✓		uses fewer general-purpose registers		✓	uses both hardwired and micro coded control unit		✓	uses a system where cache is split between data and instructions	✓			
Statement	RISC	CISC																			
uses a smaller instruction set	✓																				
uses single-cycle instructions and limited addressing modes	✓																				
uses fewer general-purpose registers		✓																			
uses both hardwired and micro coded control unit		✓																			
uses a system where cache is split between data and instructions	✓																				

Question	Answer
10(b)	<p>One mark for each correct point (<b>Max 4</b>)</p> <ul style="list-style-type: none"> <li>Instructions are divided into subtasks / 5 stages</li> <li>... Instruction fetch / IF, Instruction decode / ID, operand fetch / OF, opcode/instruction execute IE, result store / write back result / WB</li> <li>Each subtask is completed during one clock cycle</li> <li>No two instructions can execute their same stage at the same clock cycle</li> <li>The second instruction begins in the second clock cycle, while the first instruction has moved on to its second subtask.</li> <li>The third instruction begins in the third clock cycle while the first and second instructions move on to their second and third subtasks, respectively, etc.</li> </ul>

## Answer 17

Question	Answer	Marks
5	<p><b>One mark for each description (Max 2)</b></p> <ul style="list-style-type: none"> <li>• The emulation of a computer system / hardware and/or software</li> <li>• ... using a host computer system.</li> <li>• Using guest operating system(s) for emulation.</li> </ul> <p><b>One mark for each correct benefit (Max 2)</b></p> <ul style="list-style-type: none"> <li>• Multiple guest operating systems / VMs can be used on the same computer.</li> <li>• Different instruction set architectures can be emulated on a single computer.</li> <li>• A virtual machine can crash without affecting the host machine.</li> <li>• There are security benefits // Trying a piece of suspicious software and if it is / has a virus, it will only infect the virtual machine.</li> <li>• Cost savings due to not needing to purchase extra hardware.</li> <li>• Can run legacy applications that are currently incompatible.</li> </ul> <p><b>One mark for each correct drawback (Max 2)</b></p> <ul style="list-style-type: none"> <li>• A virtual machine is less efficient / has poorer performance than real machines <b>because of extra load on the host computer</b></li> <li>• Performance of the guest system cannot be adequately measured.</li> <li>• A virtual machine may be affected by any weaknesses of the host machine.</li> <li>• Costly and/or complex <b>to maintain / implement / manage</b>.</li> <li>• Cannot emulate some hardware.</li> </ul>	6

## Answer 18

Question	Answer	Marks
8	<p><b>One mark per mark point (Max 3)</b></p> <p>MP1 A <b>large number</b> of computer processors / separate computers connected together</p> <p>MP2 ... simultaneously performing a set of coordinated computations // collaborative processing</p> <p>MP3 network infrastructure</p> <p>MP4 communicate using a message interface / by sending messages.</p>	3

## Answer 19

Question	Answer	Marks
8(a)	<p><b>One mark per mark point (Max 2)</b></p> <p>MP1 Pipelining allows several <b>instructions</b> to be processed simultaneously / concurrently.</p> <p>MP2 ... therefore, increasing the CPU <b>instruction</b> throughput / the number of <b>instructions</b> completed per unit of time.</p> <p>MP3 Each <b>instruction</b> stage / subtask is completed during one clock cycle</p> <p>MP4 No two <b>instructions</b> can execute their same stage of instruction / subtask at the same clock cycle.</p> <p>MP5 ... e.g., while one <b>instruction</b> is being decoded, the next <b>instruction</b> can be fetched, etc.</p>	2

Question	Answer	Marks																																																																																																																																																																																						
8(b)	<p><b>One mark per mark point (Max 4)</b></p> <ul style="list-style-type: none"> <li>First stage of first instruction in first clock cycle</li> <li>First instruction successfully completed over five clock cycles</li> <li>Remaining instructions completed over ten clock cycles ...</li> <li>... with no extra instruction sections added on any row.</li> </ul> <p>Example answer 1</p> <table style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th colspan="13">Clock cycles</th> </tr> <tr> <th></th> <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th> <th>6</th> <th>7</th> <th>8</th> <th>9</th> <th>10</th> <th>11</th> <th>12</th> </tr> </thead> <tbody> <tr> <td>Processor stages</td> <td>IF</td> <td>1.1</td> <td>2.1</td> <td>3.1</td> <td>4.1</td> <td>5.1</td> <td>6.1</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>ID</td> <td></td> <td>1.2</td> <td>2.2</td> <td>3.2</td> <td>4.2</td> <td>5.2</td> <td>6.2</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>OF</td> <td></td> <td></td> <td>1.3</td> <td>2.3</td> <td>3.3</td> <td>4.3</td> <td>5.3</td> <td>6.3</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>IE</td> <td></td> <td></td> <td></td> <td>1.4</td> <td>2.4</td> <td>3.4</td> <td>4.4</td> <td>5.4</td> <td>6.4</td> <td></td> <td></td> </tr> <tr> <td></td> <td>WB</td> <td></td> <td></td> <td></td> <td></td> <td>1.5</td> <td>2.5</td> <td>3.5</td> <td>4.5</td> <td>5.5</td> <td>6.5</td> <td></td> </tr> </tbody> </table> <p>Example answer 2</p> <table style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th colspan="13">Clock cycles</th> </tr> <tr> <th></th> <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th> <th>6</th> <th>7</th> <th>8</th> <th>9</th> <th>10</th> <th>11</th> <th>12</th> </tr> </thead> <tbody> <tr> <td>Processor stages</td> <td>IF</td> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>ID</td> <td></td> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>OF</td> <td></td> <td></td> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>IE</td> <td></td> <td></td> <td></td> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> <td></td> <td></td> </tr> <tr> <td></td> <td>WB</td> <td></td> <td></td> <td></td> <td></td> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> <td></td> </tr> </tbody> </table>	Clock cycles														1	2	3	4	5	6	7	8	9	10	11	12	Processor stages	IF	1.1	2.1	3.1	4.1	5.1	6.1							ID		1.2	2.2	3.2	4.2	5.2	6.2						OF			1.3	2.3	3.3	4.3	5.3	6.3					IE				1.4	2.4	3.4	4.4	5.4	6.4				WB					1.5	2.5	3.5	4.5	5.5	6.5		Clock cycles														1	2	3	4	5	6	7	8	9	10	11	12	Processor stages	IF	A	B	C	D	E	F							ID		A	B	C	D	E	F						OF			A	B	C	D	E	F					IE				A	B	C	D	E	F				WB					A	B	C	D	E	F		4
Clock cycles																																																																																																																																																																																								
	1	2	3	4	5	6	7	8	9	10	11	12																																																																																																																																																																												
Processor stages	IF	1.1	2.1	3.1	4.1	5.1	6.1																																																																																																																																																																																	
	ID		1.2	2.2	3.2	4.2	5.2	6.2																																																																																																																																																																																
	OF			1.3	2.3	3.3	4.3	5.3	6.3																																																																																																																																																																															
	IE				1.4	2.4	3.4	4.4	5.4	6.4																																																																																																																																																																														
	WB					1.5	2.5	3.5	4.5	5.5	6.5																																																																																																																																																																													
Clock cycles																																																																																																																																																																																								
	1	2	3	4	5	6	7	8	9	10	11	12																																																																																																																																																																												
Processor stages	IF	A	B	C	D	E	F																																																																																																																																																																																	
	ID		A	B	C	D	E	F																																																																																																																																																																																
	OF			A	B	C	D	E	F																																																																																																																																																																															
	IE				A	B	C	D	E	F																																																																																																																																																																														
	WB					A	B	C	D	E	F																																																																																																																																																																													

## Answer 20

Question	Answer	Marks
5	<p><b>One</b> mark per mark point – SISD (<b>Max 2</b>)</p> <p>MP1 Single Instruction, Single Data (architecture). // Data is taken from a single source and a single instruction is performed on the data.</p> <p>MP2 Contains <b>one</b> processor, a control unit and a memory unit.</p> <p>MP3 ...that executes instructions sequentially.</p> <p><b>One</b> mark per mark point – MIMD (<b>Max 2</b>)</p> <p>MP4 Multiple Instruction, Multiple Data (architecture). // At any time, any processor can execute different instructions on different sets of data.</p> <p>MP5 Contains <b>many</b> processors</p> <p>MP6 ...that operate <b>asynchronously / independently</b>.</p>	4

## Answer 21

Question	Answer	Marks
10	<p><b>One</b> mark per mark point – SIMD (<b>Max 2</b>)</p> <p>MP1 Single Instruction, Multiple Data (architecture) // Performs the same operation on multiple different data streams simultaneously.</p> <p>MP2 The instructions can be performed sequentially, taking advantage of <b>pipelining</b>.</p> <p>MP3 Parallel computers <b>with</b> multiple processors.</p> <p><b>One</b> mark per mark point – MISD (<b>Max 2</b>)</p> <p>MP4 Multiple Instruction, Single Data (architecture) // Performs different operations on the same data stream.</p> <p>MP5 Each processor works on the same data stream independently.</p> <p>MP6 Parallel computers <b>with</b> multiple processors.</p>	4

## Answer 22

Question	Answer	Marks
11(a)	<p><b>One mark per mark point (Max 2)</b></p> <p>MP1 Uses hard-wired code/control units      MP2 Uses relatively few instructions / simple instructions      MP3 Uses relatively few addressing modes      MP4 Makes use of a single-cycle for each instruction      MP5 Makes use of fixed length / fixed format instructions      MP6 Makes use of general-purpose registers      MP7 Pipelining is <b>straightforward to apply</b>      MP8 The design emphasis is on the <b>software</b>      MP9 Processor chips require few transistors.</p>	2
11(b)	<p><b>One mark per mark point (Max 3)</b></p> <p>MP1 Once the processor detects an interrupt at the start/end of the fetch-execute cycle      MP2 ... the current program is temporarily stopped and the <b>status of each register</b> stored on the stack.      MP3 After the interrupt has been serviced/the Interrupt Service Routine (ISR) has been executed ...      MP4 ... the <b>registers</b> can be restored to its original status before the interrupt was detected // ... the data can be restored <b>from the stack</b>.</p>	3
11(c)	<p><b>One mark per mark point (Max 3)</b></p> <p>MP1 Pipelining adds an additional complexity // there could be a number of instructions still in the pipeline when the interrupt is received      MP2 All the instructions currently in operation are usually discarded except for the last one/the one at write back      MP3 ... the interrupt handler routine is applied to the remaining instruction.      MP4 Once the interrupt has been serviced the processor can restart with the next instruction in the sequence.</p>	3

## Answer 23

Question	Answer	Marks
9(a)	<p><b>One mark per mark point for up to two benefits (Max 2)</b></p> <p>MP1 COMPATIBILITY e.g. Applications that aren't compatible with the host computer can be run on the virtual machine // It is possible to emulate old software on a new system by running a compatible guest operating system as a virtual machine // Software can be tried on different OS on the same hardware.</p> <p>MP2 PROTECTION e.g. The guest operating system has no effect on anything outside the virtual machine other virtual machines or the host computer//Virtual machines are useful for testing as they will not crash the host computer if something goes wrong // Easier to recover if software causes a system crash as virtual machine software protects the host system.</p> <p>MP3 COST e.g. No need to buy extra computers / hardware as multiple virtual machines can be implemented on the same hardware.</p> <p><b>One mark per mark point for up to two limitations (Max 2)</b></p> <p>MP4 PERFORMANCE e.g. The performance of the guest operating system will not be as good on a virtual machine as it would be on its own compatible machine because of the extra code / using more RAM/memory space // The performance of the VM is dependent on the capabilities of the host computer // Response times cannot be accurately measured using a virtual machine.</p> <p>MP5 COMPLEXITY e.g. Building an in-house virtual machine can be expensive, time consuming and complex to maintain / set-up.</p> <p>MP6 HARDWARE/SOFTWARE ISSUES e.g. Some hardware/software can't be emulated with a virtual machine // Some of the host machine's hardware can't be directly accessed by the virtual machine.</p>	4
9(b)	<p><b>One mark per mark point – host operating system (Max 2)</b></p> <p>MP1 The host operating system is the normal operating system for the host computer / machine.      MP2 It has control of all the resources of the host computer / machine. // It can access the physical resources of the host computer / machine.      MP3 It provides a user interface to operate the virtual machine software.      MP4 It also runs the virtual machine software.</p> <p><b>One mark per mark point – guest operating system (Max 2)</b></p> <p>MP5 The guest operating system <b>runs</b> within the virtual machine.      MP6 ... it controls the virtual hardware/software during the emulation. // It accesses the actual hardware through the virtual machine and host operating system.      MP7 It provides a virtual user interface for the emulated hardware/software.      MP8 The guest operating system runs under the control of the host operating system.</p>	3