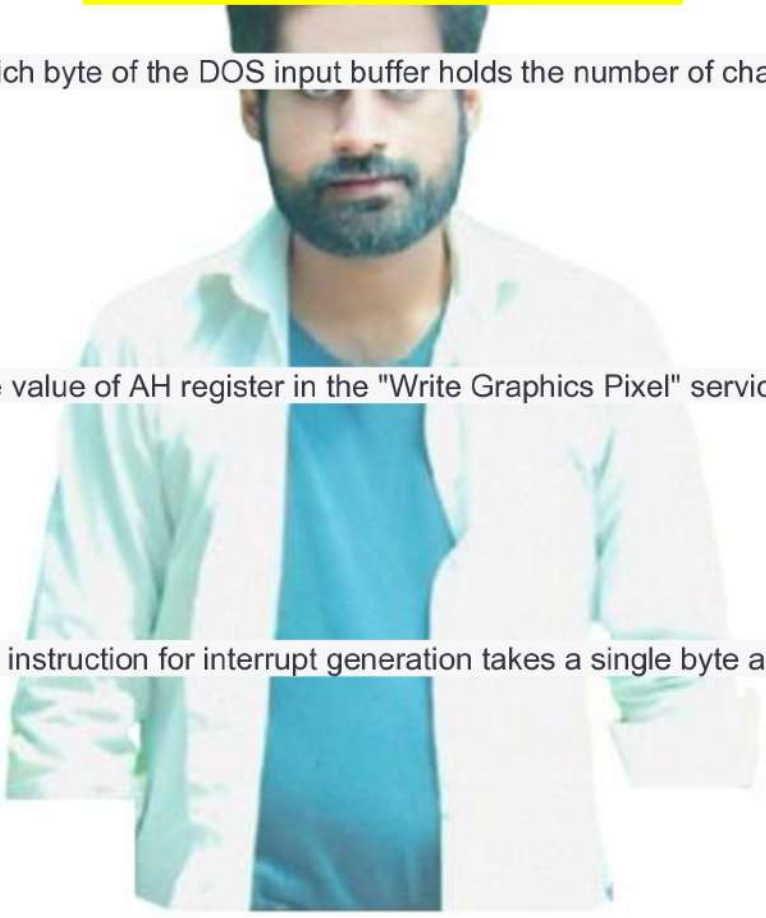
CS401 FINAL TERM PAST PAPER BY VU TEAM HADI



QUESTION: Which byte of the DOS input buffer holds the number of characters actually read on return?

First Byte
Second Byte
Third Byte
Last Byte

QUESTION: The value of AH register in the "Write Graphics Pixel" service is_____

0Ch
0Bh
1Ch
2Ch

QUESTION:INT instruction for interrupt generation takes a single byte argument varying from

0 to 127
0 to 255
-128 to 127
0 to 8

QUESTION: In Motorola 68k processors, which of the following registers can hold addresses in indirect memory access?

A0-A7
B0-B7

CO-C7
D0-D7

QUESTION: Parallel port connector is called

DB-25

BD-25
DB-24
BD-24

QUESTION: In assembly language, which of the following is used to terminate a string?

\t
S
\O
;

QUESTION: In 8086 processor, the extended form of AX register is renamed as____

AXE
EAX
AXX
XAX

QUESTION: Which of the following is a single step interrupt?

INT 0
INT 2
INT 3
INT 1

QUESTION: Which one is valid for PASCAL as a naming convention?
user_account
_Useraccount
USERACCOUNT
userAccount(check it)

QUESTION: In which register the error code is returned while using INT 21- Free Memory service?

AX
BX
CX
DX

QUESTION: Which of the following is used to clear the trap flag?

clrscr
cli
sti
Through stack

QUESTION: Port__is used to send "End of Interrupt" message to Programmable Interrupt Controller (PIC).

0x20
0x19
0x22
0x21

QUESTION:_____provides low level services as compared to_____

Device driver, DOS
DOS, BIOS
Operating system, BIOS
BIOS, operating system

QUESTION: In the original IBM XT, there was one PIC handling____possible interrupt sources.

8
4

16

2

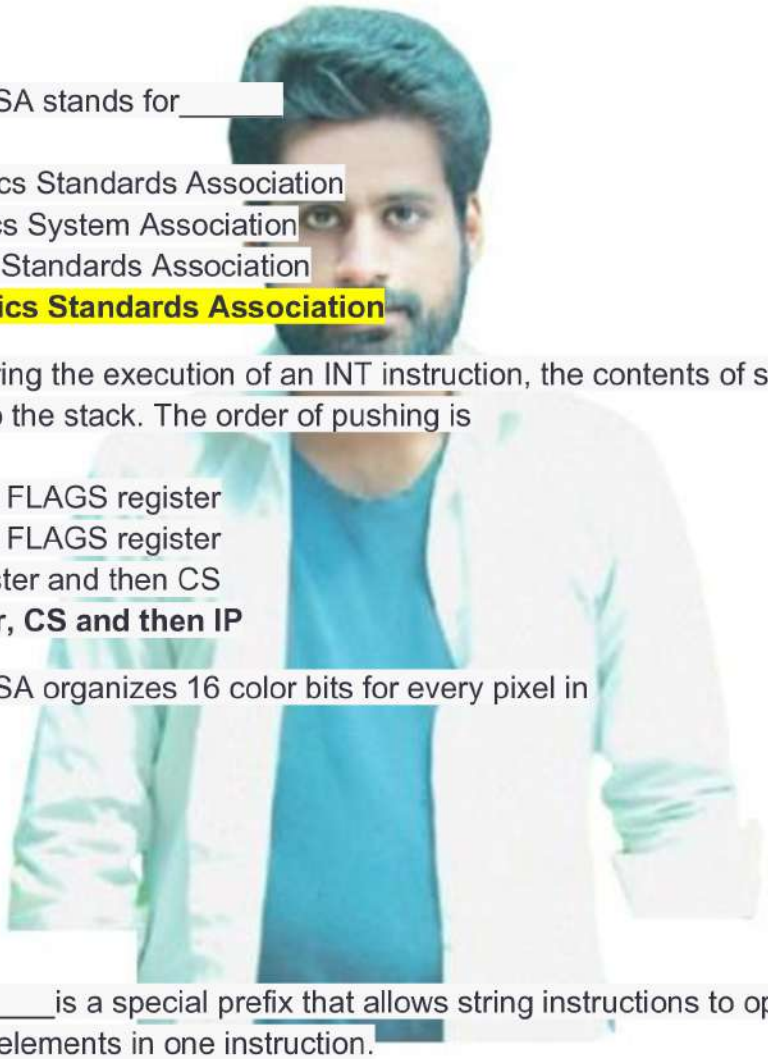
QUESTION: In protected mode, IDTR is loaded with another special instruction which is called__

GDTR
LGDT

IVT
IDT

QUESTION: There is an interval of_____ms between two timer ticks.

55
54
56
58



QUESTION:VESA stands for_____

Virtual Electronics Standards Association
Video Electronics System Association
Video Electrical Standards Association
Video Electronics Standards Association

QUESTION: During the execution of an INT instruction, the contents of some registers are pushed onto the stack. The order of pushing is
Answer
CS, IP and then FLAGS register
IP, CS and then FLAGS register
IP, FLAGS register and then CS
FLAGS register, CS and then IP

QUESTION:VESA organizes 16 color bits for every pixel in format.

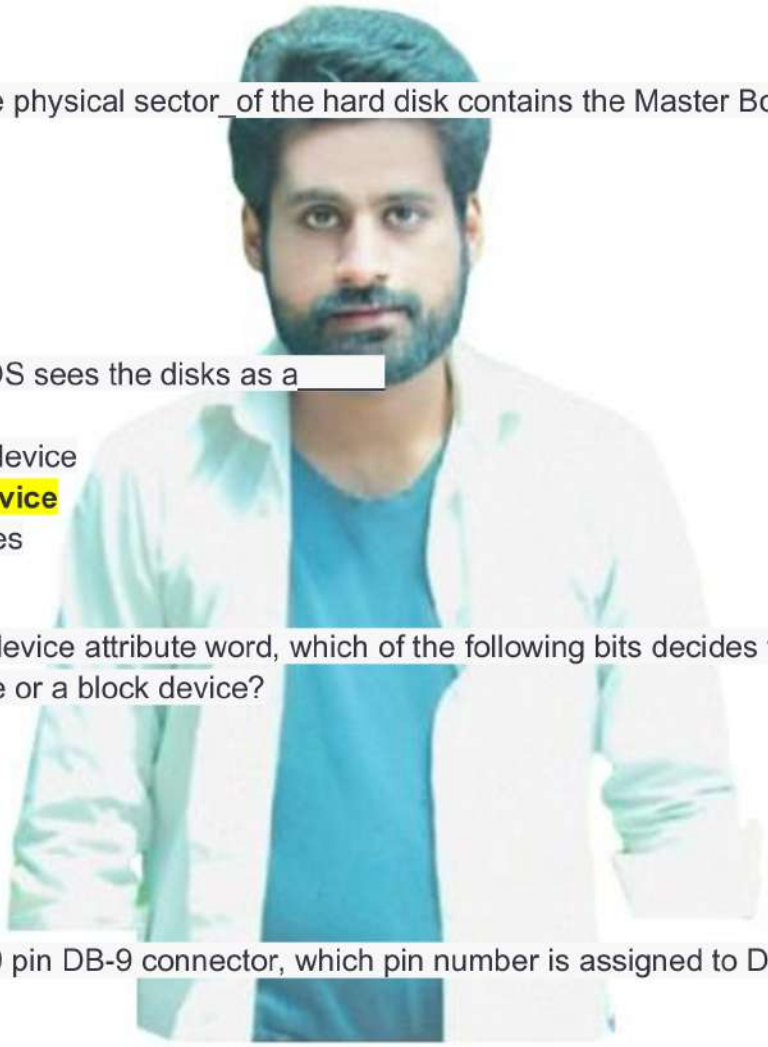
5:5:5
5:6:5
5:5:6
5:6:6

QUESTION:_____is a special prefix that allows string instructions to operate on a number of data elements in one instruction.

REPEAT
RPT
REP
REPB

QUESTION: In string manipulation,_____will hold the pointer to the destination memory.

ES:DI
ES:BP
DS:BP
DS:SI



QUESTION: The physical sector_of the hard disk contains the Master Boot Record (MBR).
0
1
2
3

QUESTION:BIOS sees the disks as a_____

logical storage device
raw storage device
series of switches
hash table

QUESTION: In device attribute word, which of the following bits decides whether it is a character device or a block device?

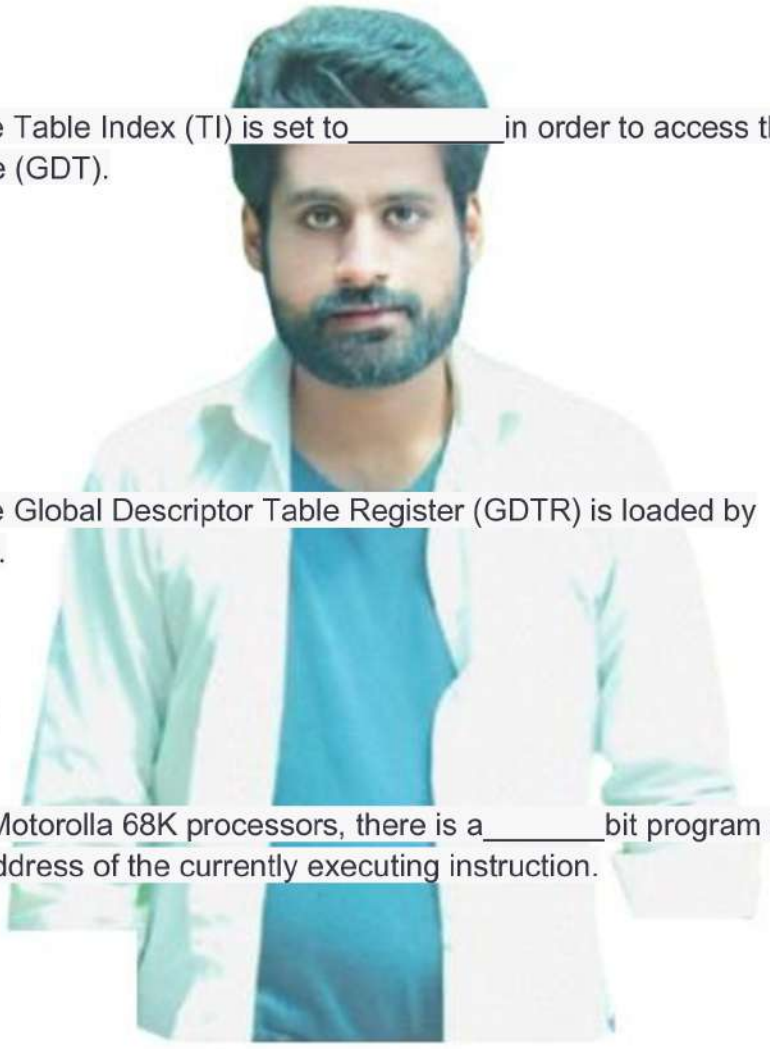
Bit 12
Bit 13
Bit 14
Bit 15

QUESTION: In 9 pin DB-9 connector, which pin number is assigned to Data Set Ready (DSR)?

O
5
9
7

QUESTION: Which of the following registers is not a part of iAPX386 architecture?

EBX
ECX
EAX
AX…check it



QUESTION: The Table Index (TI) is set to_____in order to access the Global Descriptor Table (GDT).

1
0

-1
-2

QUESTION: The Global Descriptor Table Register (GDTR) is loaded by the__instruction.

LD GDTR
LD GDT, 0x01
LD GDTR, 0x01
LGDT

QUESTION: In Motorolla 68K processors, there is a_____bit program counter (PC) that holds the address of the currently executing instruction.

8
16
32
22

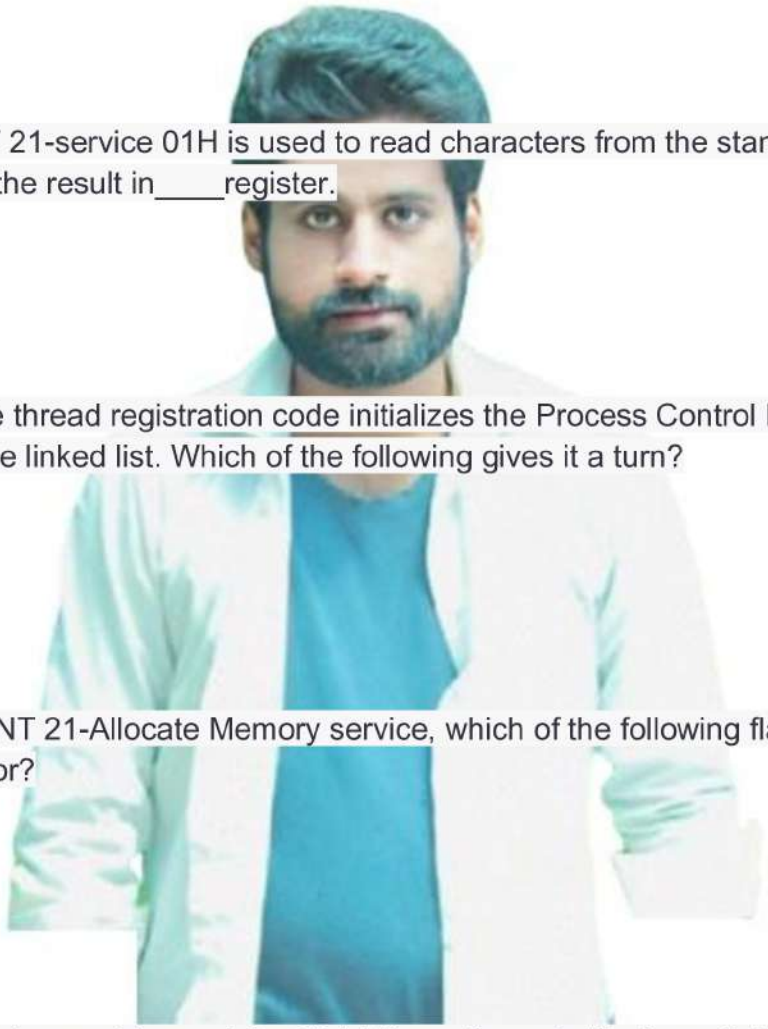
QUESTION: In computer architecture, CISC stands for___

Complex Instruction Set Computer
Complex Instruction System Computer

Correct Instruction Set Computer
Correct Instruction System Computer

QUESTION: At the end of servicing an Interrupt Service Routine (ISR), the interrupt handler should send an_____signal to the Programmable Interrupt Controller (PIC).

EI
EOI
ENDI
CLI



QUESTION:INT 21-service 01H is used to read characters from the standard input with echo. It returns the result in____register.
AL
BL
CL
BH

QUESTION: The thread registration code initializes the Process Control Block (PCB) and adds it to the linked list. Which of the following gives it a turn?

scheduler
linker
debugger
assembler

QUESTION: In INT 21-Allocate Memory service, which of the following flags is used for returning an error?

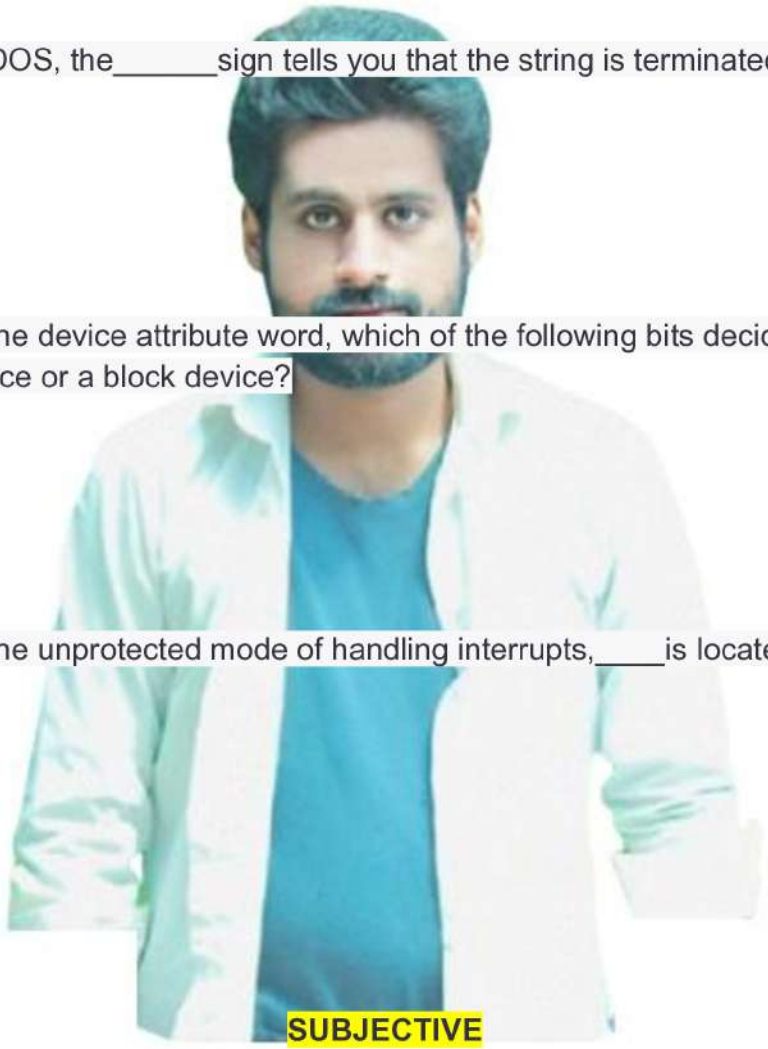
PF
CF
ZF
OF

QUESTION: The base address of the 32-bit linear frame buffer is available at offset in the mode information buffer.

26
29
27
28

The privilege level of a program varies from____(highest privilege) to ____(lowest privilege).

0,3
3,0
0,5
5,1



QUESTION: In DOS, the_____sign tells you that the string is terminated.

@
#
$
%

QUESTION: In the device attribute word, which of the following bits decides whether it is a character device or a block device?

Bit 12
Bit 13
Bit 14
Bit 15

QUESTION: In the unprotected mode of handling interrupts,____is located at physical address 0.

IVT
IDT
IDTR
GDTR

SUBJECTIVE

QUESTION: How does a processor tackle the occurrence of multiple interrupts using a single INT pin?

ANSWER:

In systems where multiple devices share a single interrupt pin, such as in older architectures or some embedded systems, a technique called "interrupt chaining" or "interrupt cascading" is often used to handle multiple interrupts.

In this approach, there is a hierarchical arrangement of interrupt controllers. The first-level interrupt controller typically handles the highest priority interrupts and is connected to the CPU's interrupt pin. When an interrupt occurs, this controller informs the CPU and temporarily masks lower-priority interrupts.

Upon receiving an interrupt signal from the first-level controller, the CPU acknowledges it and sends a signal to a second-level interrupt controller. This second-level controller, in turn, manages a set of lower-priority devices and handles their interrupts in a similar fashion. This process can continue through multiple levels of interrupt controllers, each handling a subset of devices and potentially masking lower-priority interrupts until the CPU can service them.

By using interrupt chaining, the processor can effectively manage multiple interrupts sharing a single interrupt pin while maintaining prioritization and ensuring that no interrupt is lost or overlooked.

QUESTION: Which service is provided by INT 0X21 service number 01H?

ANSWER:

INT 0x21 service number 01H provides the "Read Character with Echo" service in DOS. This service reads a single character from the standard input (usually the keyboard) and echoes it back to the screen.

QUESTION: Interrupts are said to be asynchronous. What is the meaning of asynchronous?

ANSWER:

Here's what "asynchronous" means in this context:

1. **Independent Timing**: Interrupts can occur at any time, without being synchronized to the processor's clock or the execution of specific instructions.
2. **Unpredictable Arrival**: The arrival of interrupts is not predictable in advance. They can be triggered by external events (such as hardware devices signaling completion of an operation or user input) or by software (such as a timer interrupt).
3. **Interrupt Handling**: When an interrupt occurs, the processor temporarily suspends its current activities, saves the current state, and transfers control to an interrupt handler

(also known as an Interrupt Service Routine or ISR) to process the interrupt. After handling the interrupt, the processor resumes its previous activities.

4. **Concurrency**: As interrupts can occur asynchronously, multiple interrupts may be pending or processed simultaneously, leading to concurrent execution of interrupt handlers.

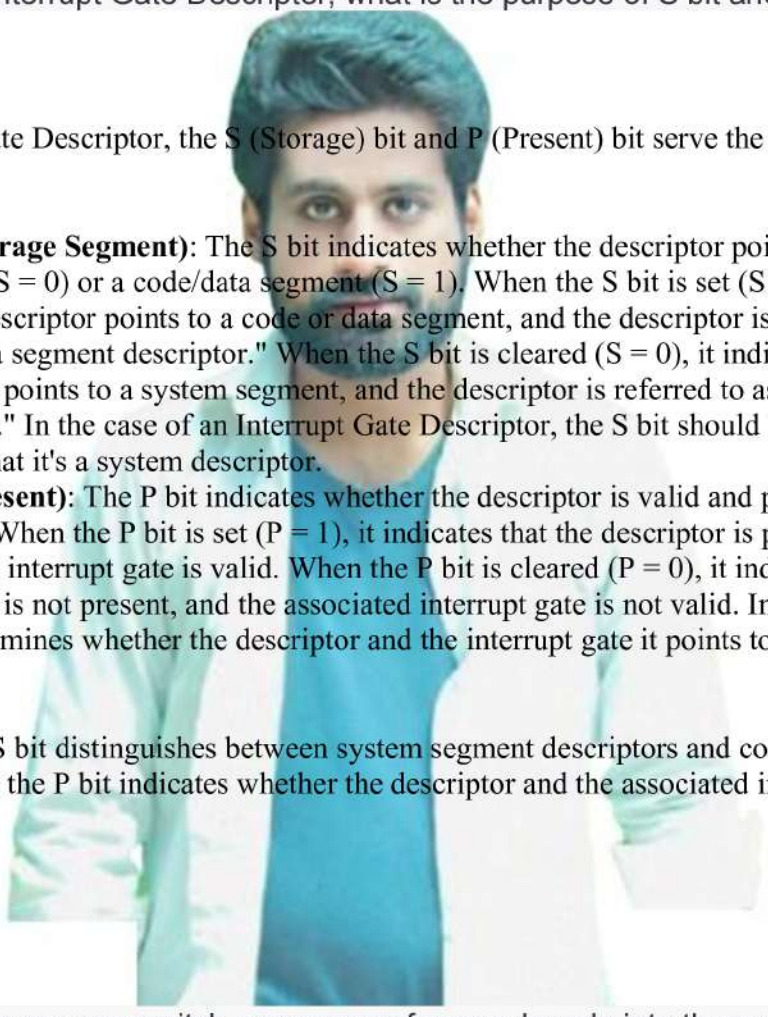
QUESTION: In Interrupt Gate Descriptor, what is the purpose of S bit and P bit?

ANSWER:

In an Interrupt Gate Descriptor, the S (Storage) bit and P (Present) bit serve the following purposes:

1. **S Bit (Storage Segment)**: The S bit indicates whether the descriptor points to a system segment (S = 0) or a code/data segment (S = 1). When the S bit is set (S = 1), it indicates that the descriptor points to a code or data segment, and the descriptor is referred to as a "code/data segment descriptor." When the S bit is cleared (S = 0), it indicates that the descriptor points to a system segment, and the descriptor is referred to as a "system descriptor." In the case of an Interrupt Gate Descriptor, the S bit should be set to 0 to indicate that it's a system descriptor.
2. **P Bit (Present)**: The P bit indicates whether the descriptor is valid and present in memory. When the P bit is set (P = 1), it indicates that the descriptor is present, and the associated interrupt gate is valid. When the P bit is cleared (P = 0), it indicates that the descriptor is not present, and the associated interrupt gate is not valid. In other words, the P bit determines whether the descriptor and the interrupt gate it points to are currently in use.

In summary, the S bit distinguishes between system segment descriptors and code/data segment descriptors, while the P bit indicates whether the descriptor and the associated interrupt gate are present and valid.

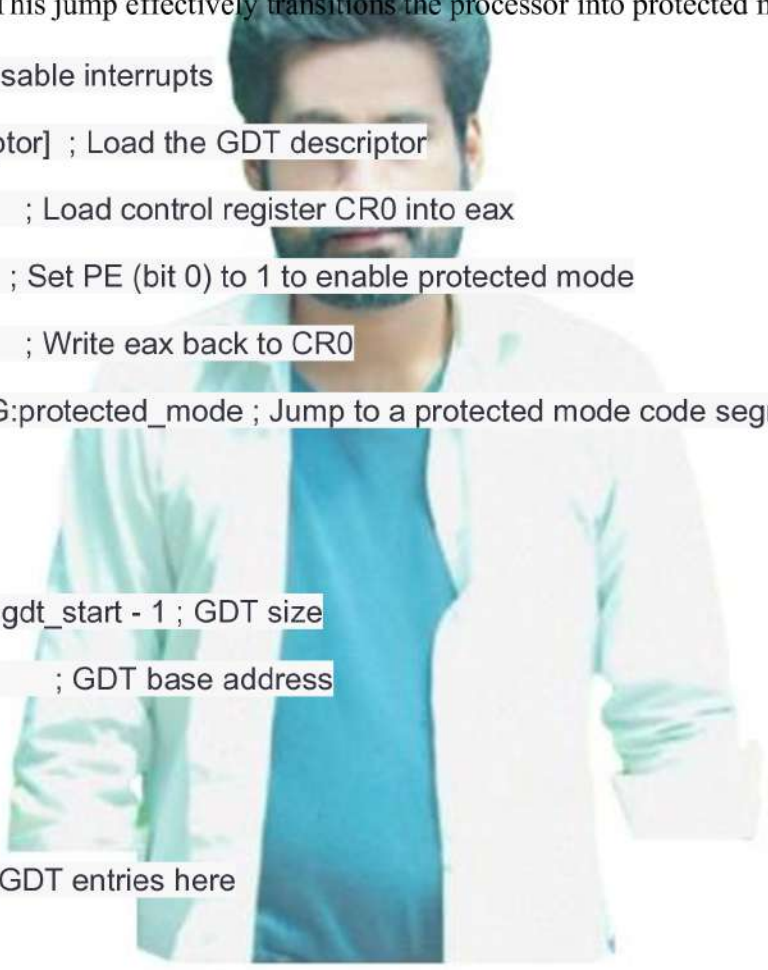


QUESTION: How can you switch a processor from real mode into the protected mode?

ANSWER:

Switching a processor from real mode into protected mode involves a series of steps:

1. **Enable the A20 line (optional)**: The A20 line must be enabled to allow access to memory beyond the 1 MB limit. This is typically done by setting appropriate bits in the keyboard controller, although modern CPUs often handle this automatically.
2. **Set up the Global Descriptor Table (GDT)**: In protected mode, memory access is controlled by segment descriptors stored in the GDT. You need to set up a GDT that describes the memory layout of the system.
3. **Load the GDT descriptor**: Load the address and size of the GDT into the GDTR register using the LGDT instruction.
4. **Set the PE bit**: The PE (Protection Enable) bit in the control register CR0 must be set to 1 to enter protected mode.
5. **Jump to a protected mode code segment**: After setting up the GDT and enabling protected mode, jump to a code segment that is defined in the GDT as a 32-bit code segment. This jump effectively transitions the processor into protected mode.



```
cli                 ; Disable interrupts

lgdt [gdt_descriptor]  ; Load the GDT descriptor

mov eax, cr0        ; Load control register CR0 into eax

or eax, 0x1         ; Set PE (bit 0) to 1 to enable protected mode

mov cr0, eax        ; Write eax back to CR0

jmp CODE_SEG:protected_mode ; Jump to a protected mode code segment


gdt_descriptor:
    dw gdt_end - gdt_start - 1 ; GDT size
    dd gdt_start        ; GDT base address


gdt_start:
    ; Define your GDT entries here

CODE_SEG equ 08h  ; Selector for a 32-bit code segment


protected_mode:
    ; Code to execute in protected mode goes here
```

QUESTION: Write the name of the highest priority interrupt.

ANSWER:

The highest priority interrupt in x86 systems is the Non-Maskable Interrupt (NMI). It is a special type of interrupt that cannot be disabled or ignored by software and is typically used for critical system events, such as hardware malfunctions or watchdog timer expirations.

What does PIC stand for in Computer Architecture? Explain the purpose and functionality of a PIC in interrupts.

ANSWER:

he purpose of a PIC is to efficiently handle interrupts generated by hardware devices such as keyboards, mice, disk drives, and network adapters. When a hardware device needs attention from the CPU, it sends an interrupt request to the PIC.

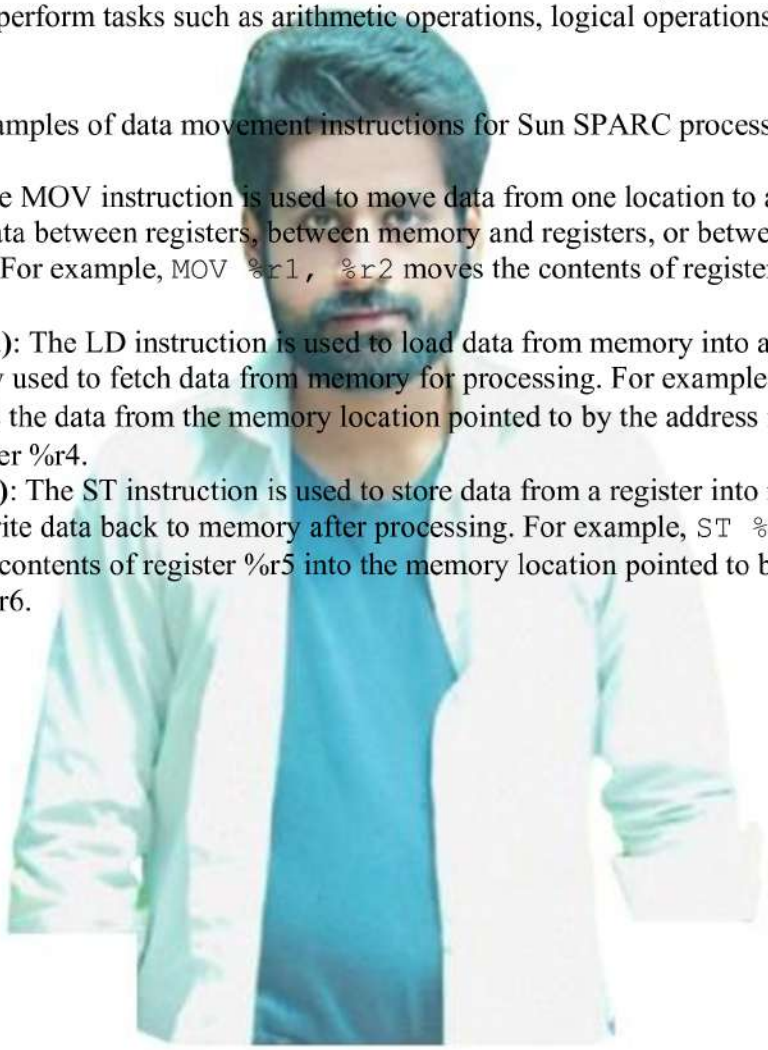
The functionality of a PIC involves several key aspects:

1. **Interrupt Prioritization**: The PIC prioritizes interrupt requests based on their importance and urgency. Higher priority interrupts are serviced before lower priority ones.
2. **Interrupt Masking**: The PIC allows interrupts to be masked or temporarily disabled. This prevents lower priority interrupts from interrupting the CPU while it is servicing a higher priority interrupt.
3. **Interrupt Vectoring**: When an interrupt request is received, the PIC determines the appropriate interrupt vector, which is a unique identifier for the interrupt. The PIC then forwards this vector to the CPU, allowing the CPU to locate and execute the corresponding interrupt service routine (ISR).
4. **Interrupt Acknowledgment**: Once the CPU has finished servicing an interrupt, it sends an acknowledgment signal (End of Interrupt or EOI) to the PIC. This signal informs the PIC that the interrupt has been handled and allows the PIC to clear the interrupt request and potentially service other pending interrupts.
5. **Cascading**: Some systems may use multiple PICs cascaded together to handle a larger number of interrupts. In such cases, one PIC acts as the master controller, while the others act as slave controllers. The master PIC communicates with the CPU, while the slave PICs handle interrupts from peripheral devices.

QUESTION: What is a processor? Provide any three examples of data movement instructions for SUN SPARC processors.

ANSWER:

A processor, also known as a central processing unit (CPU), is the primary component of a computer system responsible for executing instructions, performing calculations, and controlling the overall operation of the system. It fetches instructions from memory, decodes them, and executes them to perform tasks such as arithmetic operations, logical operations, data movement, and control flow.



Here are three examples of data movement instructions for Sun SPARC processors:

1. **MOV**: The MOV instruction is used to move data from one location to another. It can transfer data between registers, between memory and registers, or between memory locations. For example, `MOV %r1, %r2` moves the contents of register %r1 to register %r2.
2. **LD (Load)**: The LD instruction is used to load data from memory into a register. It is commonly used to fetch data from memory for processing. For example, `LD [%r3], %r4` loads the data from the memory location pointed to by the address in register %r3 into register %r4.
3. **ST (Store)**: The ST instruction is used to store data from a register into memory. It is used to write data back to memory after processing. For example, `ST %r5, [%r6]` stores the contents of register %r5 into the memory location pointed to by the address in register %r6.