What happens when 'LDA s' is run?

1

What happens when 'STA s' is run?

2

What happens when 'ADD s' is run?

3

What happens when 'SUB s' is run?

4

What happens when 'JMP s' is run?

5

What happens when 'JGE s' is run?

6

What happens when 'JNE s' is run?

7

What three steps occur during the fetch phase?

8

ACC = [s]

1

[s] = ACC

2

PC = s

3

ACC += [s]

4

PC = s

5

if ACC != 0 then PC = s

6

ACC -= [s]

7

if ACC ¿= 0 then PC = s

8

1. *Use PC as address to read memory*
2. *Save result of read in CPU*
3. *Increment PCI read*

What control signals do all registers need?

9

What control signal does a multiplexer need?

10

What control signals does the memory need?

11

Which 3 signals control the ALU?

12

What is a process?

13

What is the address space?

14

What is a thread?

15

What is multi-threading?

16

A signal to select an input

10

An enable signal

9

add, sub & byp

12

Ren (read enable) and Wen (write enable)

11

All memory locations the process can use.

14

A program in execution, the thread + address space.

13

This is where we have multiple threads within the same process

16

A sequence of instructions that are obeyed.

15

How do we make programs think they have sole use of memory?

17

What are the three different approaches to engineering an OS?

18

What are the three process states?

19

In the diagram, what is happening at each stage?

20

What is a PCB table?

21

In scheduling, what do the following mean?

1. *CPU burst*
2. *I/O burst*
3. *CPU bound*
4. *I/O bound*

22

What is a processes turnaround time?

23

What is a processes waiting time?

24

Monolithic, layered and micro-kernels.

18

*Use **relocation**, where we swap a program out of memory and later swap it back in.*

17

1. *Process need to wait for I/O or event.*
2. *Process forcibly preempted - **interrupt / relinquish CPU / time-slice expired**.*
3. *Scheduler selects process to run.*
4. *I/O or event occurs.*

20

Running, ready, blocked

19

1. *Process executing on CPU*
2. *Process blocked, waiting for I/O*
3. *Long CPU bursts*
4. *Short CPU bursts*

22

Process control block, it contains all of the information needed about processes.

21

The time that the process waits to run.

24

The time from a process being submitted to it getting completed.

23

Briefly explain the first come first served scheduling algorithm.

25

What is meant by pre-emptive scheduling?

26

What is meant by non-pre-emptive scheduling?

27

What is the fixed time amount called in non-pre-emptive processing?

28

Briefly explain the shortest remaining time first scheduling algorithm.

29

What is process starvation?

30

In scheduling, what are static priorities?

31

In scheduling, what are dynamic priorities?

32

Scheduling where processes run until they are terminated or blocked.

26

The first process in the ready state gets CPU time first. Once it is blocked or complete, the next process in the queue is run. Processes that require CPU time are added to the back of the queue.

25

The 'time-slice' or 'time-quantum'.

28

Scheduling where a process can run for some fixed maximum time, once it has reached its maximum time, it is interrupted and set 'ready' and the scheduler runs the next process.

27

When the scheduling algorithm leaves a process out for a long time, causing the process to not receive any CPU time.

30

For each newly ready process, if CPU-burst is less than the time to complete the running process then context-switch and run the new process.

29

Priorities that are assigned by the system to achieve certain goals.

32

Priorities that are predetermined for each process.

31

What do the following terms mean?

1. *Data inconsistency*
2. *Synchronisation*
3. *Critical section*
4. *Mutual exclusion*

33

What is deadlock?

34

What is the base register of a program?

35

What is the limit register of a program?

36

What is the base register usage sequence?

37

What is the limit register usage sequence?

38

What is the virtual address?

39

What performs the virtual to physical address conversion?

40

1. *Disagreement about data values*
2. *Using appropriate policies and mechanisms to ensure the correct operation of cooperating processes*
3. *Section of code in which shared data is used*
4. *At most 1 process can be in its critical section at once*

33

Where there are a set of waiting processes where each process is waiting for something that can only be provided by another of the processes.

34

A register that is loaded with the physical address where the program begins in memory.

35

A register that is loaded with the length of the program.

36

When the processor references memory, either fetch an instruction or read or write a data word, the CPU hardware automatically adds the base balue to the address generated by the processor before sending the address out on the memory bus.

37

When the base register usage sequence happens, the OS checks if the address offered is greater than the value in the limits register, in which case a fault is generated and access aborted.

38

An address that is generated by a program. It is converted to the actual 'physical address' which is used in memory.

39

The memory management unit (MMU)

40

What is the difference between a partition and a program?

41

What are the two main reasons for virtual memory in a computer system?

42

What is a page table?

43

What are the two techniques used to implement virtual memory?

44

Segmented memory is split into chunks of a _____ *size.*

45

What is the formula to work out the number of pages used in a system?

46

What is the format of an address to paged memory?

47

What steps does the MMU take when it receives a paged address?

48

1. *To allow a processor to address a much larger address space than is implemented by the physical memory*
2. *To support the OS in the management of processes*

42

A partition is: division of the storage area of a memory.
A program is: supplies a computer with a set of pre-written instructions.

41

Paged virtual memory and segmented virtual memory.

44

A table used by the MMU to translate from a virtual to a physical address.

43

$$\#pages = \frac{addressSpace}{PageSize}$$

Where addressSpace and pageSize are measured in bits.

46

Segmented memory is split into chunks of a variable size.

45

1. *Look up the page number in the page table and see if it's in physical memory.*
2. *If it is in memory, generate a physical address ((pagesize × pagenumber) + offset), and request that from RAM.*
3. *If it's not, abort the memory access, OS will load the page into memory (this is a page fault).*

48

Page number + offset

47

What columns are present in the page table?

49

In a page table, the resident flag checks whether the page is

50

In a page table the used flag check whether the page

51

In a page table, what does the dirty flag do?

52

LRU is a page replacement algorithm, what does LRU stand for?

53

FIFO is a page replacement algorithm, what does FIFO stand for?

54

- *Resident flag*
- *Used flag*
- *Dirty flag*
- *Physical address*
- *Disk address*

49

In a page table, the resident flag checks whether the page is currently in virtual memory

50

In a page table the used flag check whether the page has been used

51

Checks whether the page has been written to while it's been in physical memory (so it needs to be copied back in full when it's put back onto the disk).

52

Last recently used

53

First in first out.

54