Hello and Welcome Unit 2.

Processes: process description and control, module name,

process control module #8 presented Mr. Girish Abhyankar,,

Associate Professor, Dnyanprassarak Mandal's college and

Research Center. Assagao, Bardez Goa. Outline: modes of

execution, Process creation, Process switching.

Learning outcomes: the student will be able to explain the

modes of execution.

Explain the steps in process creation and process switching

modes of execution.

Most processors support at least two modes of execution.

The user mode and the kernel

mode. The user mode is a less privileged mode and most of the

user programs typically execute in this mode.

The kernel mode is the more privileged mode is also referred

to as system more or control mode. It is necessary to protect

the OS and key operating system tables such as process control

blocks from interference by

user programs. In kernel more the software has complete

control of the processor and all its instructions, registers &

memory. Such a level of control is not necessary and for safety

is not desirable for user programs. How does the processor know in which mode it is to be executing? either user mode or Kernel? Typically there is a bit in program status word PSW that indicates the mode of execution.

This bit is changed in response to certain events.

When a user makes a call to an OS service or when an interrupt triggers execution of an operating system routine,

the mode is set to the kernel mode and upon return from the service to the user process, the mode is set to user mode.

So this is how it is distinguished between the kernel mode and a user mode.

process creation: in the previous module we have seen what are the reasons for creating a process in an OS.

Let us now look what are the steps once it is decided to process, create a process. What are the steps to be followed?

Step number one.

Assign a unique process identifier to the new process.

At this time, a new entry is added to the process table's primary process table, which contains one entry per process.

So at the start we give an ID to a process whenever it is created, step #2.

Allocate space for the process in the memory of course

this includes all elements of the process image.

What is process image? You may recollect it is a user program,

user data stack and PCB together it becomes a process

image. Thus, the OS must know how much space is needed

for a private user address space, thats program and data,

and the user stack.

Third step, initialize the PCB.That's a process control block.

Which will have basically three types of information. One is

process identification, Processor state information and

process control information.

Step #4, set the appropriate linkages. For example, if the OS

maintains each scheduling queue as a linked list, then the new

process must be put in the ready or ready suspend list.

Step #5, create or expand other data structures. What do I mean by

that? For example, the OS may maintain an accounting file on

each process to be used subsequently for billing and or

Performance assessment purposes so these are

basically the five steps increating a process.

Process switching.

At some time, a running process is

Interrupted and the OS assigns another process to

the running state and turns control over to that process.

This is process switching. When to switch a process or. When

does a process which happens? A process which may occur anytime

that the OS has gained control from the currently running

process. Possible mechanisms.

Interrupt, trap, supervisor call.

Interrupt, which is due to some sort of event that is

external to and independent of the currently running

process, such as completion of an I/O operation.

Trap it relates to an order, an error, or exception condition

generated within the currently running process, such as an

illegal file access attempt.

So trap is the one which is error within the running

program. Interrupt is something which comes outside from the

running program supervisor call from a program being executed.

So let us look at the interrupts little bit in detail. We have

different types of interrupts: Clock interrupt: when the

currently running process has been executed for a maximum

allowable limit of time, we call it as a time slice.

Then the process must be switched to a ready state and

another process is to be dispatched for running.
I/O interrupt the OS determines what I/O action has

occurred. If the input output action constitutes an event for

which one or more processes are waiting, then the OS moves all

of the corresponding blocked processes to ready state and

blocked suspend processes to ready suspend state.

The OS then decides whether to resume execution of the process

currently in the running state,or to preempt that process for a

higher priority ready process.

Memory fault: the processor encounters a virtual memory

address reference for a word which is not in the main memory.

Then the particular block which contains that word from the

secondary memory has to be brought to the main memory.

So after the I/O request is issued to bring in the block of

memory, the process with memory fault is placed in a blocked

state. The OS then performs a process switch to resume

execution of another process.

After the desired block is brought into the memory, the

process is placed in the ready state, which is this process.

The process that process which was suspended as it was not

having the required word in the

main memory. WHen to speed the process another reason

trapped. The OS determines if the error or exception condition is fatal. If so, then the currently running processes move to exit state and the process which occurs. remember trap occurs within the executive program. If not, then the action of the OS will depend on the nature of the error and the design of the operating system.

It may attempt some recovery procedure or simply notify the user. It may do a process switch or resume the currently running process. This is in case of a trap supervisor call. The OS may be activated by a supervisor call from the program being executed. For example, a user process is running and an instruction is executed that requests an I/O operation, such as file open.

This call results in a transfer to a routine that is a part of OS code. The use of system call may place the user process in the block state, because unless that happens, this cannot proceed. Change of process state: What exactly is the change of process? How exactly the change of process happens?

Step number one. Save the context of the processor, including the program counter,and other registers.

Update the PCB of the process that is currently

running because now the state will change from running It

will go to any of the other state ready suspend block

depending upon the situation.So thus you have to update the

process control block.

Other relevant fields also must be updated, including the

reasons for leaving the running state and the accounting

information. The next step is to move the PCB of this process to

the appropriate Q, whether it has been moved to ready, then it

should go to the ready queue. If it is blocked on an event, it

should go to that, etc.

Step #4 select another process for execution. Once we remove

this process away from the processor, the processor

cannot be left idle. So here to bring in other process. step

#5 update the PCB of the process selected.

This includes changing the state of that selected process to be

running. Step 6.

Update the memory management data structures. This may

require. This may be required depending on how address

translation is managed. step #7, restore the context

of the processor to that which existed at the time the
Selected Newly selected process was last switched

Out of th running state by loading in the previous values of the program counter

and the other registers.

Please remember if this is a totally new process then the

Context the reloading of the previously saved context

will not happen.

This is as per the references shown here.

Thank you.