ENG 428

Interrupt service routine to make a function run with a higher priority.

non-deterministic = random delay.

RTX has deterministic delay - RTOS  
Cortex - no cache, just main memory

PROCESS/TASK - instance of computer program being executed = job to be done by CPU  
- written in c (for us)  
- when process is created, allocates resources including: process stack, memory address space, registers, PL etc (this is the process control block)  
- It's data is not shared with other tasks/processes  
inactive = waiting to be scheduled

MULTI-TASKING - Method that allows multiple processes to share the CPU  
- OS maximizes CPU usage by interleaving processes  
- Processes can be assigned priority, used for scheduling

Can be in 1 of 4 states at a given time  
1)Running 2)Ready 3)Blocking - cant be exe until certain event has occurred 4)INACTIVE - waiting to be scheduled

KERNEL - computer program that acts as an interface between application and hardware  
- Manages I/O requests from application, translates into instructions for CPU  
- Manages, synchronizes, & provides communication between tasks

2 Spaces = KERNEL SPACE - handles exe of processes & handles interrupts  
USER SPACE - running applications

System call - when a process makes a request to kernel to execute a service it doesn't have permission to run.  
Critical region - section of code that cannot be interrupted by another process.  
Mutex - control access to critical regions/resources for multiple processes (acts as a lock)  
Semaphore - a mutex that can be shared amongst n tasks/processes versus 1. (4 keys for example)

RTOS - only has a kernel (it is the os) Reg OS has a kernel and other.

|  |  |  |
| --- | --- | --- |
| Application | | |
| KERNEL | | |
| CPU | MEM | DEVICES |

Thermostat - embedded system for multitasking: sense, change, display, schedule.

RT Systems - system w/ specific response time requirements  
Soft RT - deadlines are important, but will still function correctly if deadlines are occasionally missed.  
- tasks may have improved priorities than other tasks. - data acquisition

Firm RT - Soft RT but do not benefit from late service delivery  
- constant failures lead to system degradation - Pacemaker, Seismic detector, circuit board creation

Hard RT - system that absolutely must meet its deadline  
- performance not met = system failed - emergency stop? flight control system

Real RT - Hard RT systems that have very short response times. - nuclear reactor, lasers,

Round Robin

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 4 |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |  |
|  | 0 | 10 | 20 | 30 | 40 | 50 | 60 |  |  |

Preemptive (interruptible)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Ta | Ts |
| Task | Priority | Arrival | T exe |
| A | 255 | 7 | 10 |
| B | 50 | 5 | 10 |
| C | 10 | 0 | 10 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| B |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| A |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 17 | 25 | 30 |

Non-Preemptive

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C |  |  |  |  |  |  |  |  |  |
| B |  |  |  |  |  |  |  |  |  |
| A |  |  |  |  |  |  |  |  |  |
|  | 0 | 10 | 20 | 30 |  |  |  |  |  |

API - application interface - built in functions, software and microcontroller. Header

ROUNDROBIN - time sliced scheduling, gives poor response times for RT systems, but appropriate for multi-programming environments (equal priorities)  
= IO bound - processes use CPU for minimal time & requests/executes an I/O if available. Blocks I/O for other processes until complete. therefore other tasks are also blocked until task is done with I/O

FCFS(first come first served) - process that has been waiting longest in ready queue is sent for execution, process runs to completion while other tasks are waiting.  
Tf = finish time; Tr = Tf-Ta = Response time

|  |  |  |
| --- | --- | --- |
| Process | Ta | Ts (service) |
| 1 | 0 | 3 |
| 2 | 2 | 6 |
| 3 | 4 | 4 |
| 4 | 6 | 5 |
| 5 | 8 | 2 |

RR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |

FCFS

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| P5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process | A | B | C | D |
| Ta | 0 | 1 | 2 | 3 |
| Ts | 1 | 9 | 1 | 9 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PCFS | A | B | B | B | B | B | B | B | B | B | C | D | D | D | D | D | D | ... |
| RR | A | B | C | B | D | B | D | B | D | B | D | B | D | B | D | B | D | ... |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| FCFS | **A** | **B** | **C** | **D** | RR | **A** | **B** | **C** | **D** |
| **Tf** | 1 | 10 | 11 | 20 | **Tf** | 1 | 18 | 3 | 20 |
| **Tr** | 1 | 9 | 9 | 17 | **Tr** | 1 | 17 | 1 | 17 |
| **Tr/Ts** | 1/1 | 9/9 | 9 | 1.89 | **Tr/Ts** | 1 | 1.89 | 1 | 1.89 |

Homework:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E |
| Ta | 0 | 1 | 3 | 9 | 12 |
| Ts | 3 | 5 | 2 | 5 | 5 |

FCFS Tf, Tr, Tr/Ts Queued first, Running Second  
RR q=1  
RR q=4