Module: Dispatcher

Extends: Thread

Field Variables

|  |  |  |
| --- | --- | --- |
| Field Name | Type | Meaning |
| processSelected | Process | Represents the process that is to be sent to the CPU |
| cpu | CPU | Represents a CPU process |

INTERFACE:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Method Name | Input | Output | Description | State Transitions | Dependencies |
| Dispatcher |  | Dispatcher | Constructor | INITIAL STATE:  processSelected=null |  |
| run | void | void | Overrided method from Thread class  The dispatcher class continually tries to select a Process from the static ReadyQueue object generated by the RRScheduler class.  The Dispatcher's progress is regulated by the ReadyQueue object. (It is blocked and notified by said monitor on when it is appropriate to select a Process)  After it selects a Process, the Dispatcher allows the CPU process  The Dispatcher process then gives the selected Process to the CPU thread and starts it- then it waits to finish by using the  The dispatcher generates a message each time processSelected changes. | See select | this.select()  Thread.join() |

IMPLEMENTATION:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Method Name | Input | Output | Description | State Transitions | Dependencies |
| select | void | void | take a Process from the RRScheduler.readyQueue  Generate a message indicating a Process was selected | processSelected=RRScheduler.readyQ.select()  cpu=new CPU(5000)  cpu.load(processSelected)  cpu.start()  cpu.join() | ReadyQueue.select()  CPU.CPU  CPU.load()  Thread.start()  Thread.join() |
| generateMessage() |  | none | Generates a message of the form  "Process # loaded into CPU) |  |  |