FileReader	Processor		Process
- file: fstream - pams: parameters[]	- id_count: Int - preeption_counter: Int - scheduler: Scheduler*		- id: Int - creation_date: Int
+ readFile(): Void	- process_list: Vector <process*> - old_process: Int</process*>	<b>~</b>	- start_date: Int - end_date: Int - period: Int
	- active_process: Int - SP: Uint64_t - PC: Uint64_t - ST: Uint64_t - REG: Vector <uint64_t> - context_change_counter: Int - time_val: time_t - time_counter: Int - max_instances: Int - instances: Vector<int> - deadline_loss: vector<int> + run(): Void - loadProcess(vector<parameters*> pams): Void</parameters*></int></int></uint64_t>		- period. Int - old_deadline: Int - deadline: Int - deadline: Int - duration: Int - priority: Int - status: String - current_executed_time: Int - total_executed_time: Int - wait_time: Int - turn_around_time: Vector <int> - wait_time_vector: Vector<int> - SP: Uint64_t - PC: Uint64_t - ST: Uint64_t</int></int>
	- printContext(): Void - changeContext(int old_process, int active_process):		- REG: Vector <uint64_t></uint64_t>
	Void - checkComputionTIme(): Void - checkDeadline(): Void - updateActiveProcess(int active_process): Void - updateAbsDeadlines(): Void - chooseAlgorithm(): Void - chooseNumInstatnces(): Void - printStatus(): Void - printTimes(): Void		- generateContext(): Void + getId(): Int + getStartDate(): Int + getEndDate(): Int + getPeriod(): Int + getDeadline(): Int + getAbsDeadline(): Int + getOldDeadline(): Int + getDuration(): Int + getPriority(): Int + getPriority(): Int + getStatus(): String + getCurrentExecutedTime(): Int + getTotalExecutedTime(): Int + getWaitTime(): Int + getWaitTimeVector(): Vector <int> + getTurnAroundTime(): Vector<int> + getSP(): Uint64_t + getPC(): Uint64_t</int></int>
	Scheduler		+ getST(): Uint64_t + getREG(): Vector <uint64_t></uint64_t>
	- processID_vector: Vector <int> - process_list: Vector<process*> - algorithm: Int</process*></int>		+ setStartDate(int v): Void + setEndDate(int v): Void + setDeadLine(int v): Void + setAbsDeadLine(int v): Void
	+ defineAlgorithm(int a): Void + getActiveProcess(): Void + schedule(Vector <process*> process_list_): Void - sortVector(bool (Schedule::*func)(Int, Int)): Void - sort(bool (Schedule::*func)(Int, Int)): Void - compareByAbsDeadline(Int a, Int b): Bool - compareByPriority(Int a, Int b): Bool</process*>		+ setOldDeadLine(int v): Void + setStatus(string v): Void + setCurrentExecutedTime(int v): Void + setTotalExecutedTime(int v): Void + setWaitTime(int v): Void + setSP(uint64_t v): Void + setPC(uint64_t v): Void + setST(uint64_t v): Void + setREG(vector <uint64_t> v): Void + setTurnAroundTime(vector<int> v): Void + setWaitTimeVector(vector<int> v): Void</int></int></uint64_t>