

## **INE5412: Sistemas Operacionais 1**

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

### **Trabalho 1**

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

Professor Giovani Gracioli

1 de setembro de 2023

### **Estrutura**

Resumo

1. Estrutura final
2. Organização do código
3. Dificuldades
4. Referências

## **INE5412: Sistemas Operacionais 1**

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

### **Resumo**

A dupla apresenta a abordagem adotada na implementação de uma abstração da execução de escalonamento de um conjunto de processos por meio da utilização de variadas políticas ou algoritmos de escalonamento. Serão detalhadas questões de organização de código, dificuldades enfrentadas ao longo da implementação e o diagrama usado como base para a produção do trabalho.

## **INE5412: Sistemas Operacionais 1**

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

### **1. Estrutura Final**

Na imagem a seguir, é possível observar o diagrama UML produzido para a execução do trabalho:

Diagrama UML:

[https://lucid.app/lucidchart/37cea13e-1669-40ee-8ed3-c1718278e0b6/edit?viewport\\_loc=-349%2C-113%2C2964%2C1255%2CHWEp-vi-RSFO&invitationId=inv\\_f5509ea9-b50e-4bb3-b17f-3c18d99e6511](https://lucid.app/lucidchart/37cea13e-1669-40ee-8ed3-c1718278e0b6/edit?viewport_loc=-349%2C-113%2C2964%2C1255%2CHWEp-vi-RSFO&invitationId=inv_f5509ea9-b50e-4bb3-b17f-3c18d99e6511)

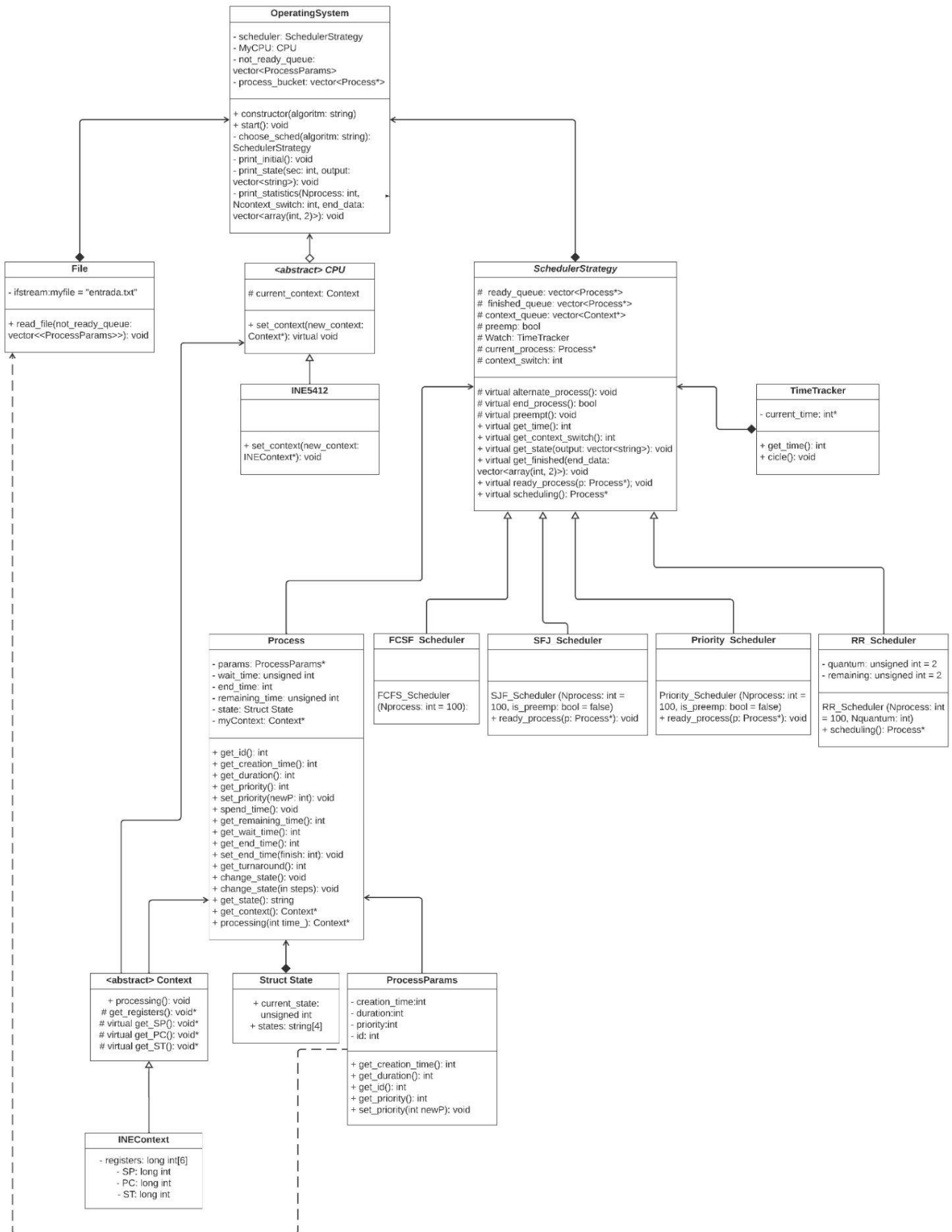
*Imagem 1 - diagrama UML final*

Também é possível acessar o diagrama pelo arquivo disponibilizado pelo Moodle.

# INE5412: Sistemas Operacionais 1

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199



### 2. Organização do código

- **Operating system:** ao executar o arquivo main, o algoritmo de escalonamento é selecionado e uma instância do **sistema operacional** é criada. O sistema operacional é responsável por realizar a leitura do arquivo de entrada e repassar os processos para o algoritmo de escalonamento correspondente. Ele também terá como atributo uma instância da CPU, classe abstrata implementada de forma a possibilitar a fácil adaptação para diferentes modelos;

- **File:** a classe File faz a leitura do arquivo “entrada.txt” e cria uma lista (ordenada pela data de criação) com os parâmetros de processos lidos;

- **CPU:** a classe abstrata CPU detém o contexto e os métodos que realizam a alteração do contexto dos processos. A CPU específica utilizada é a INE5412, que herda a classe abstrata;

- **Scheduler:** o Scheduler é responsável por ter a fila de processos prontos e fazer o controle de ordem e tempo, modelado pela classe **TimeTracker**, durante a execução. É também responsável por preparar os processos para serem adicionados na fila de prontos de cada algoritmo de escalonamento, além de decidir quando e como um processo será preemptado.

- SchedulerStrategy é uma classe abstrata que implementa vários métodos virtuais que suas instâncias (FCFS\_Scheduler, SJF\_Scheduler, Priority\_Scheduler e RR\_Scheduler) sobrescrever apenas quando necessário. Por exemplo, FCFS\_Scheduler não sobrescreve nenhum método (apenas cria seu próprio construtor), enquanto SJF\_Scheduler e Priority\_Scheduler sobrescrevem ready\_process() (que insere um processo na fila de prontos de acordo com o critério do algoritmo) e preempt() (que define como será a preempção). RR\_Scheduler é o único que precisa sobrescrever o método scheduling().

## INE5412: Sistemas Operacionais 1

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

- **Process:** a classe Process é responsável por modelar os processos, salvando seus parâmetros, tempo de espera, tempo de execução, tempo restante de execução e contexto, além do estado. Estado é uma instância da struct **State**;

- **ProcessParams:** controla o ID e os parâmetros de tempo e prioridade do processo;

- **Context:** classe abstrata que contém os registradores de propósito geral e também os registradores especiais: program counter, stack pointer e status. Também é responsável por, na sua inicialização e na execução de algum processo, simular a escrita nos registradores. Nesse caso, são escritos valores aleatórios nos registradores. Sendo uma classe abstrata, as funções getters dos registradores usam ponteiros void, enquanto a implementação INEcontext usa long int (para emular um processador de 64 bits).

De forma resumida, ao criar uma instância do sistema operacional, é criada uma classe CPU INE5412, cujo contexto é do tipo INEcontext, com registradores de propósito geral, PC, SP e ST. Para simular uma operação com salvamento de informações, um valor aleatório é adicionado nos registradores a cada execução. O sistema operacional também usa a fila de parâmetros dos processos lidos do arquivo de entrada (que, portanto, não estão prontos ainda) para criar os processos quando determinado pelo escalonador. Esses processos serão então passados ao escalonador para definir sua execução pelo algoritmo de escolha do usuário.

Assim, o Scheduler possuirá uma fila de processos prontos e processos finalizados e criará uma instância da classe TimeTracker, que faz o controle do tempo atual e do clock, definido em 1 segundo. Então, cada algoritmo de escalonamento obtém sua fila e realiza a computação necessária para a execução dos processos.

## **INE5412: Sistemas Operacionais 1**

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

### **Dificuldades**

As dificuldades enfrentadas na realização deste trabalho envolveram principalmente o que diz respeito à organização da estrutura de classes, devido ao grau de complexidade apresentado pelo trabalho. Assim, o diagrama UML utilizado passou por diversas alterações ao longo do projeto, envolvendo criações, deleções e alterações de classes. Para ilustrar, segue a imagem de uma das versões intermediárias do UML:

# INE5412: Sistemas Operacionais 1

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

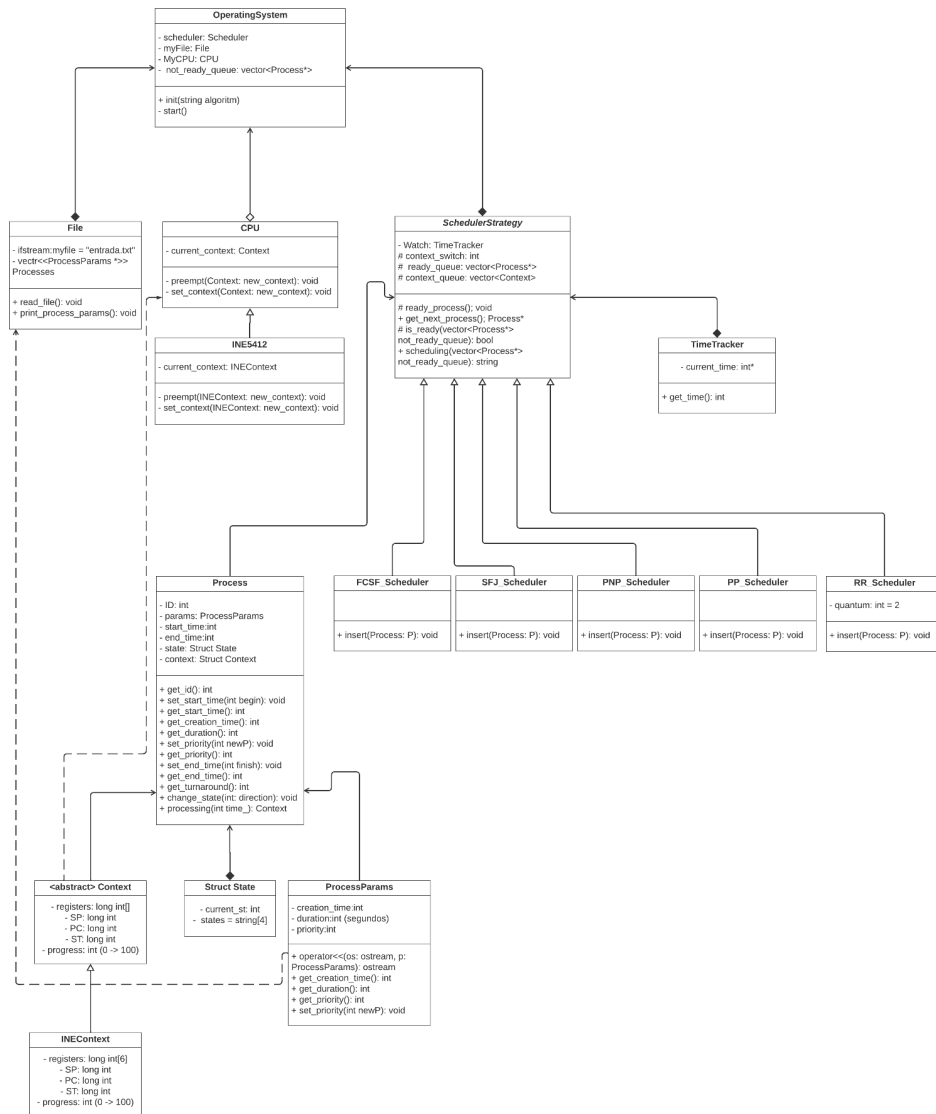


Imagem 2 - imagem do progresso intermediário do diagrama UML

Como é possível perceber pela imagem, o método que realizava a preempção, presente na CPU INE5412, deixou de existir na versão final.



**INE5412: Sistemas Operacionais 1**

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

Muitas outras mudanças foram feitas ao longo da execução do trabalho, como a criação de um Scheduler para cada política de escalonamento ao invés de uma única classe de scheduler para vários algoritmos. Assim, cada um dos algoritmos de escalonamento é responsável por gerenciar o processo em execução, a troca de contexto e o controle de tempo.

## INE5412: Sistemas Operacionais 1

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

### Especificações

O programa foi testado e encontrado funcional para:

- Sistema Operacional: Linux-gnu x86\_64, Ubuntu 23.04
- Compilador: g++ (Ubuntu 12.3.0-1ubuntu1~23.04) 12.3.0
  - Configured with: `../src/configure -v --with-pkgversion='Ubuntu 12.3.0-1ubuntu1~23.04' --with-bugurl=file:///usr/share/doc/gcc-12/README.Bugs --enable-languages=c,ada,c++,go,d,fortran,objc,obj-c++,m2 --prefix=/usr --with-gcc-major-version-only --program-suffix=-12 --program-prefix=x86_64-linux-gnu- --enable-shared --enable-linker-build-id --libexecdir=/usr/lib --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --enable-clocale=gnu --enable-libstdcxx-debug --enable-libstdcxx-time=yes --with-default-libstdcxx-abi=new --enable-gnu-unique-object --disable-vtable-verify --enable-plugin --enable-default-pie --with-system-zlib --enable-libphobos-checking=release --with-target-system-zlib=auto --enable-objc-gc=auto --enable-multiarch --disable-werror --enable-cet --with-arch-32=i686 --with-abi=m64 --with-multilib-list=m32,m64,mx32 --enable-multilib --with-tune=generic --enable-offload-targets=nvptx-none=/build/gcc-12-DAPbBt/gcc-12-12.3.0/debian/tmp-nvptx/usr,amdgc-n-amdhsa=/build/gcc-12-DAPbBt/gcc-12-12.3.0/debian/tmp-gcn/usr --enable-offload-defaulted --without-cuda-driver --enable-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu`

## **INE5412: Sistemas Operacionais 1**

Clara Rosa Oliveira Gonçalves, matrícula 22103511

Maykon Marcos Junior, matrícula 22102199

- IDE: VS Code 1.82.2

### **Referências**

- Materiais da disciplina.