

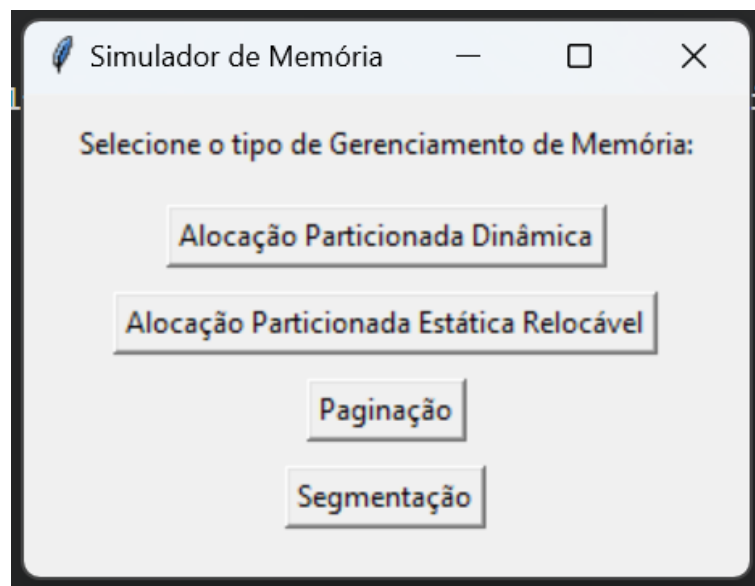
## Trabalho Final - Sistemas Operacionais

**Estudantes:** Nicole Fernanda Rodrigues Alves e Thais Jandre

### Simulador de Gerenciamento de Memória

O presente trabalho, desenvolvido utilizando Python como linguagem de programação, tem como objetivo simular os diferentes tipos de Gerenciamento de Memória. Para melhor visualização, o sistema possui uma interface gráfica que permite ao usuário escolher o tipo de gerenciamento de memória que deseja testar, além de permitir a visualização dos processos sendo alocados e desalocados da memória conforme desejado.

Para inicializar o sistema, basta executar o arquivo 'apresentacao.py' e a seguinte tela será apresentada:



Conforme mencionado anteriormente, a tela inicial permite ao usuário escolher uma das quatro variações de gerenciamento de memória: Alocação Particionada Dinâmica, Alocação Particionada Estática Relocável, Paginação ou Segmentação. Vale destacar que em cada uma das quatro variações, o usuário terá disponíveis dois campos de input e três botões, especificados abaixo:

#### Campos de Input:

- ID do Processo: campo no qual o usuário deverá indicar qual o número do ID que irá identificar o processo a ser alocado ou desalocado da memória.
- Tamanho do Processo: campo a ser utilizado na alocação de processos, indicando qual o tamanho daquele processo em especial.

#### Botões:

- Alocar: ao preencher ambos os campos e clicar no botão alocar, o sistema ocupa parte da memória com aquele processo. Ao fazer isso, a parte na qual está alocado o processo muda de cor para rosa.

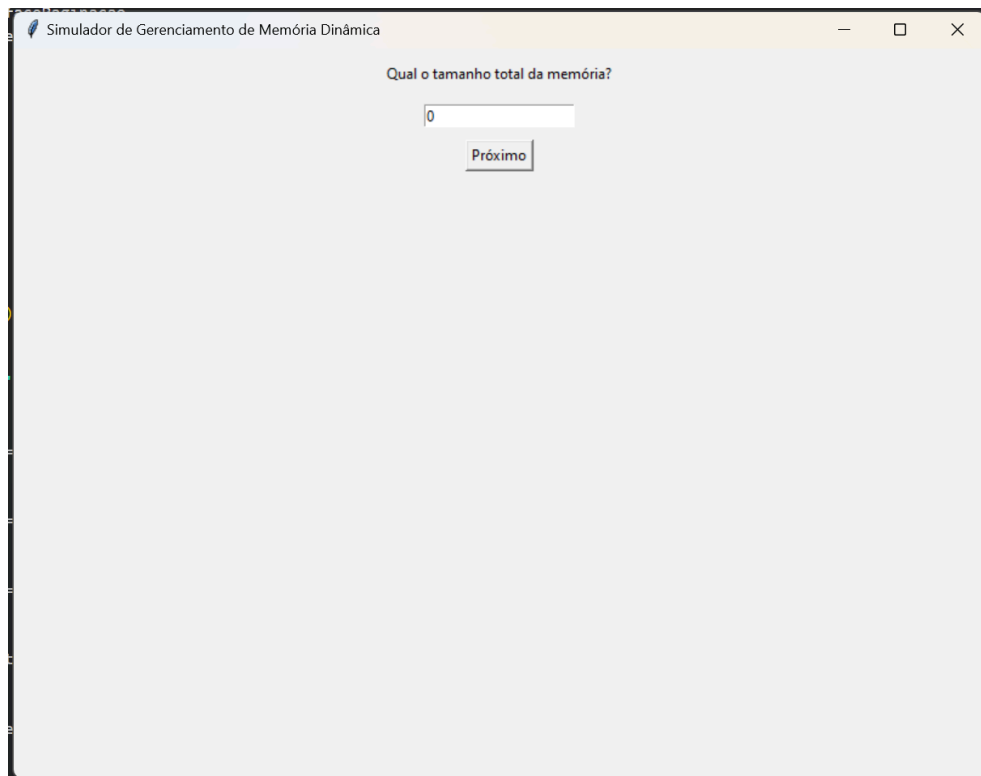
- Desalocar: ao identificar o número de ID do processo desejado e clicar em desalocar, o sistema remove aquele processo em específico da memória e a cor do espaço ocupado pelo processo volta a ser azul.
- Limpar: ao clicar nesse botão, todos os processos são desalocados da memória e ela volta a ter seu espaço total livre e, conseqüentemente, identificado na cor azul.

O sistema conta ainda com pop-ups de mensagem de sucesso e erro, tanto para alocação quanto para a desalocação de processos em memória e também para a limpeza da memória. Uma vez que os pop-ups se repetem para as quatro variações de gerenciamento, eles serão mostrados apenas na primeira variação apresentada.

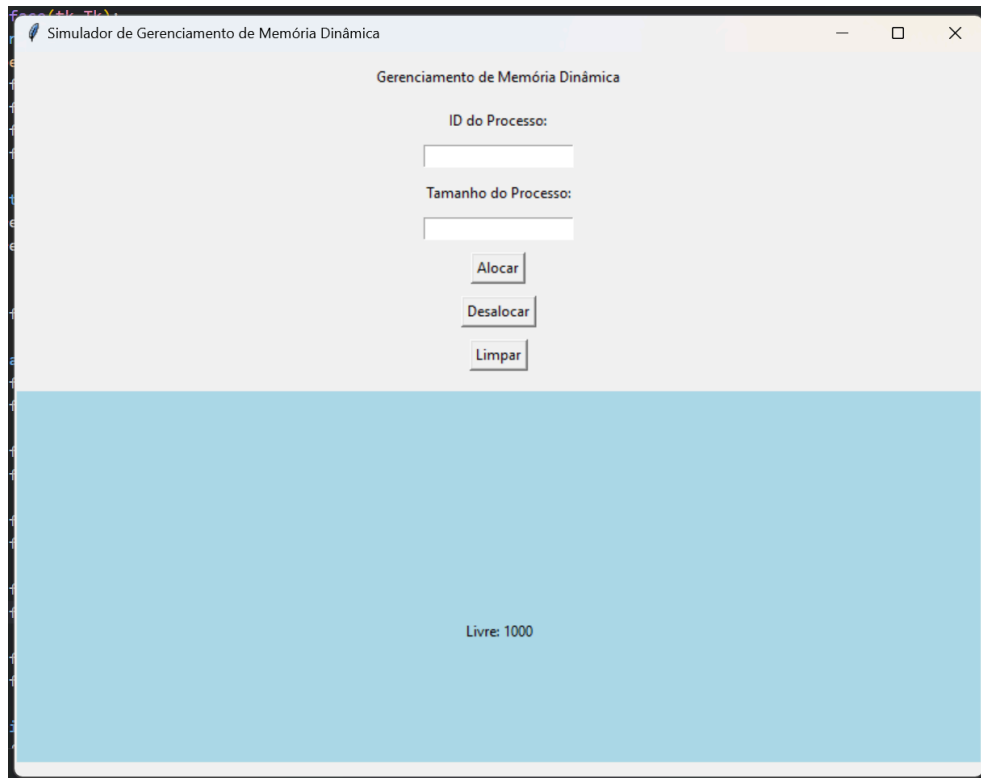
Por fim, vale lembrar que para simular um tipo diferente de gerenciamento de memória, o usuário deve fechar a tela de exibição da interface gráfica e executar novamente o arquivo 'apresentacao.py'. No presente relatório, vamos passar por cada uma das variações e seu funcionamento ao alocar e desalocar processos.

### **Alocação Particionada Dinâmica**

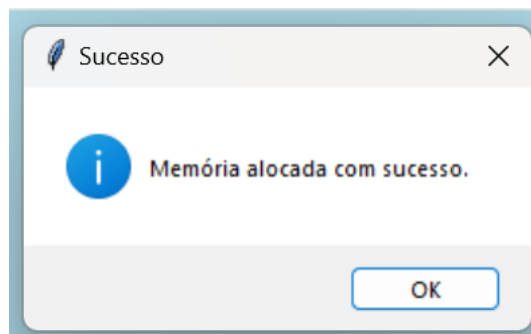
Ao escolher a primeira opção, a tela abaixo será apresentada, na qual o usuário deverá indicar qual o tamanho total desejado para a memória e então clicar no botão 'Próximo'. Como exemplo, vamos usar 1000.

A imagem mostra uma janela de aplicativo com o título "Simulador de Gerenciamento de Memória Dinâmica". O conteúdo da janela é simples, com o texto "Qual o tamanho total da memória?" no topo. Abaixo dele, há um campo de entrada de texto contendo o número "0". Logo abaixo do campo, há um botão com o rótulo "Próximo". A janela possui uma barra de título padrão com ícones de minimizar, maximizar e fechar.

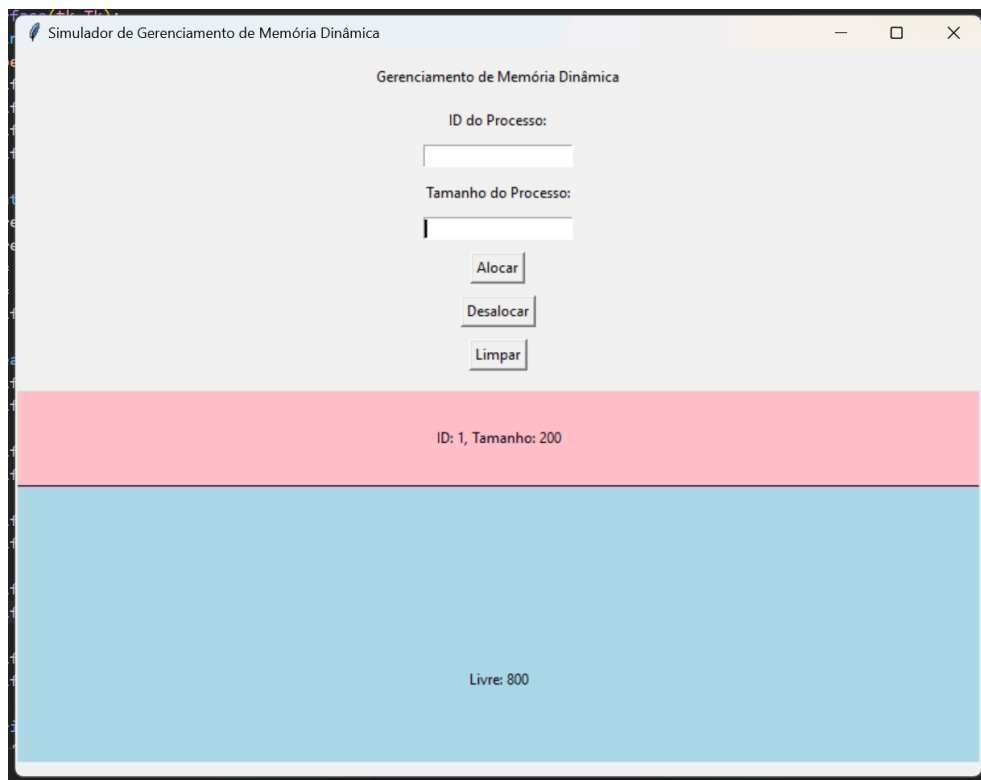
Então, tem-se a seguinte tela, com o tamanho total livre indicado em azul.



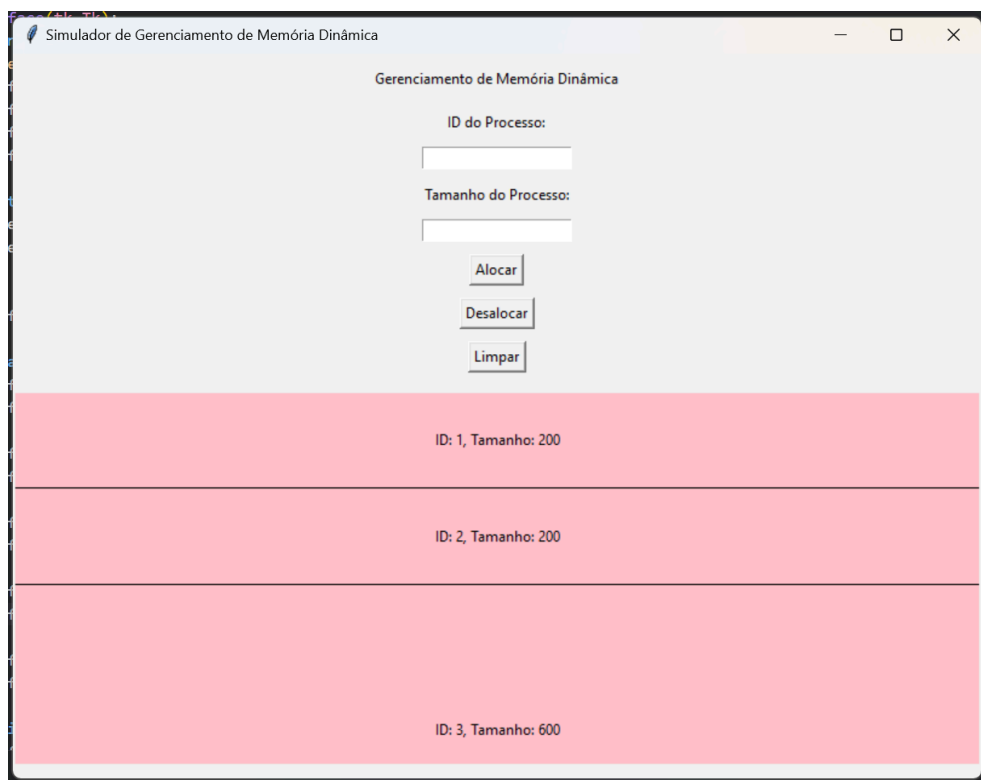
Ao alocar com sucesso um processo de ID 1 e tamanho 200, primeiramente tem-se uma mensagem de sucesso, exibida em um pop-up:



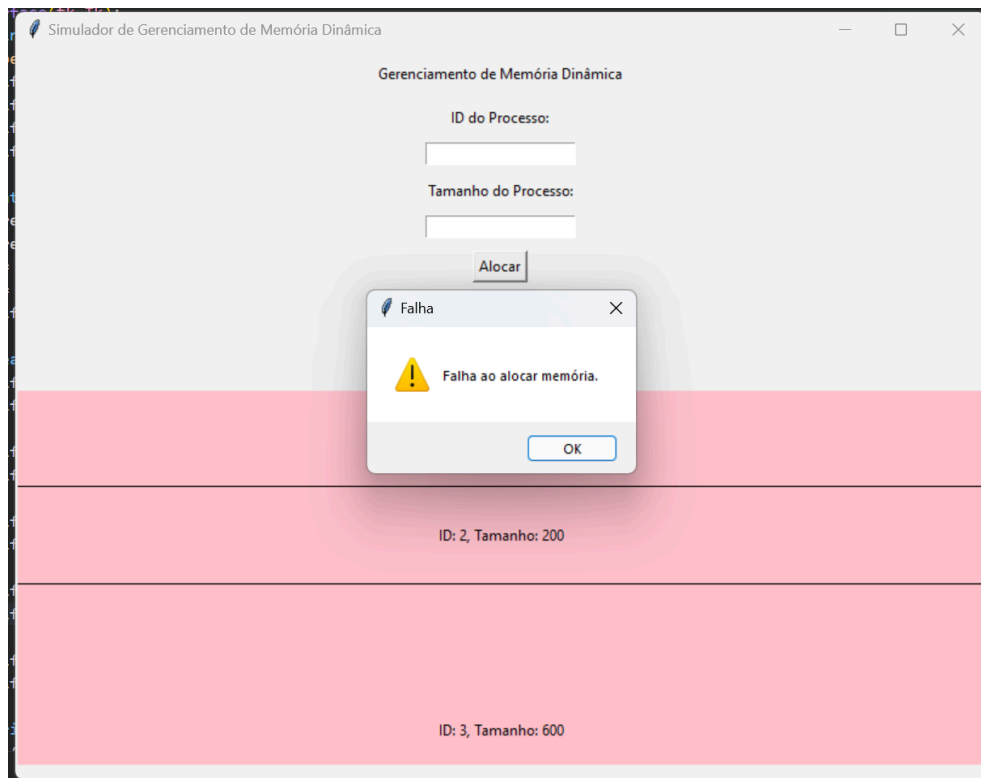
Dando OK no pop-up, tem-se:



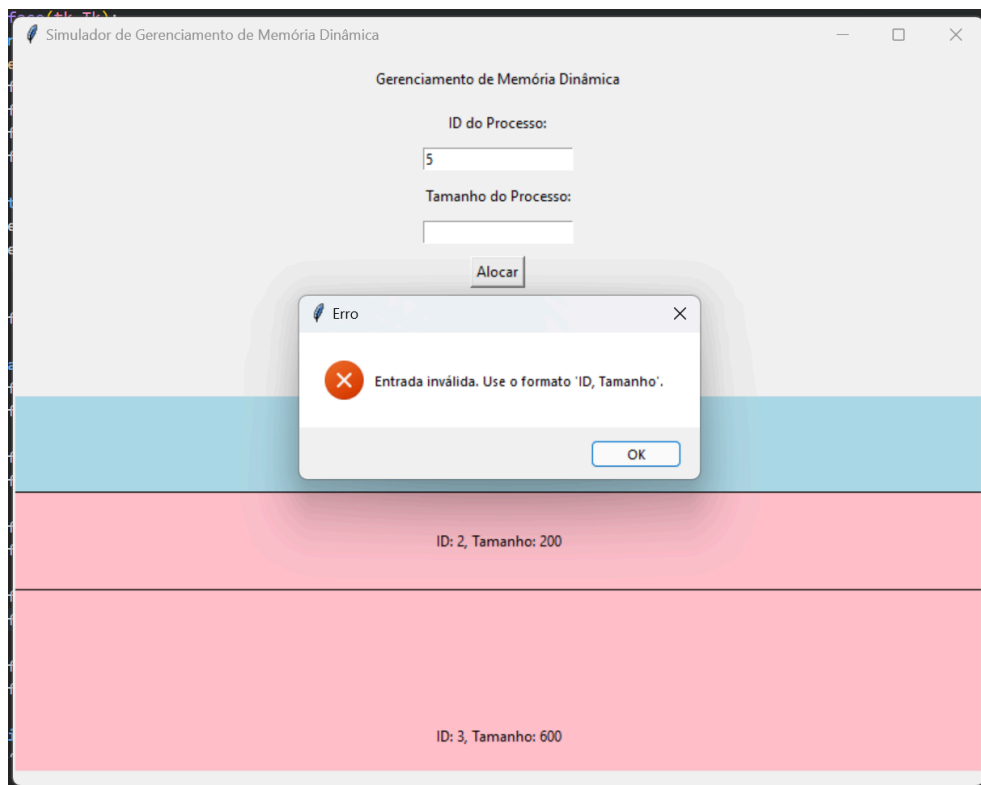
Uma vez tendo preenchido a memória totalmente, tem-se:



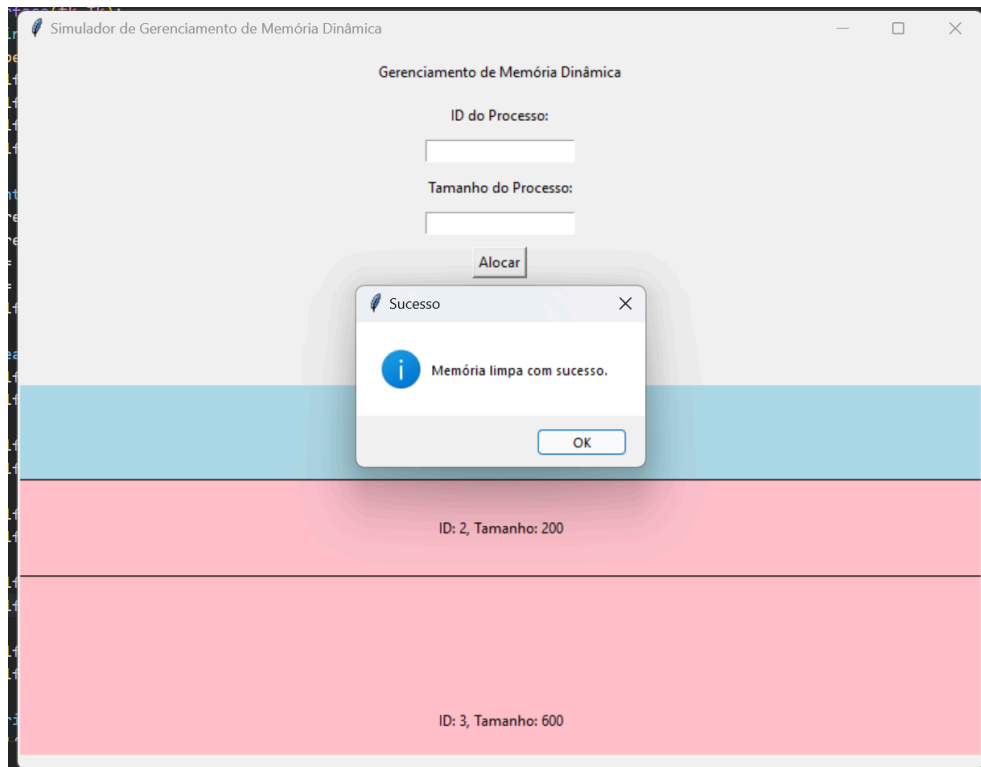
Nesse caso, ao tentar alocar mais um processo obtemos uma falha de alocação na memória, já que ela está cheia.



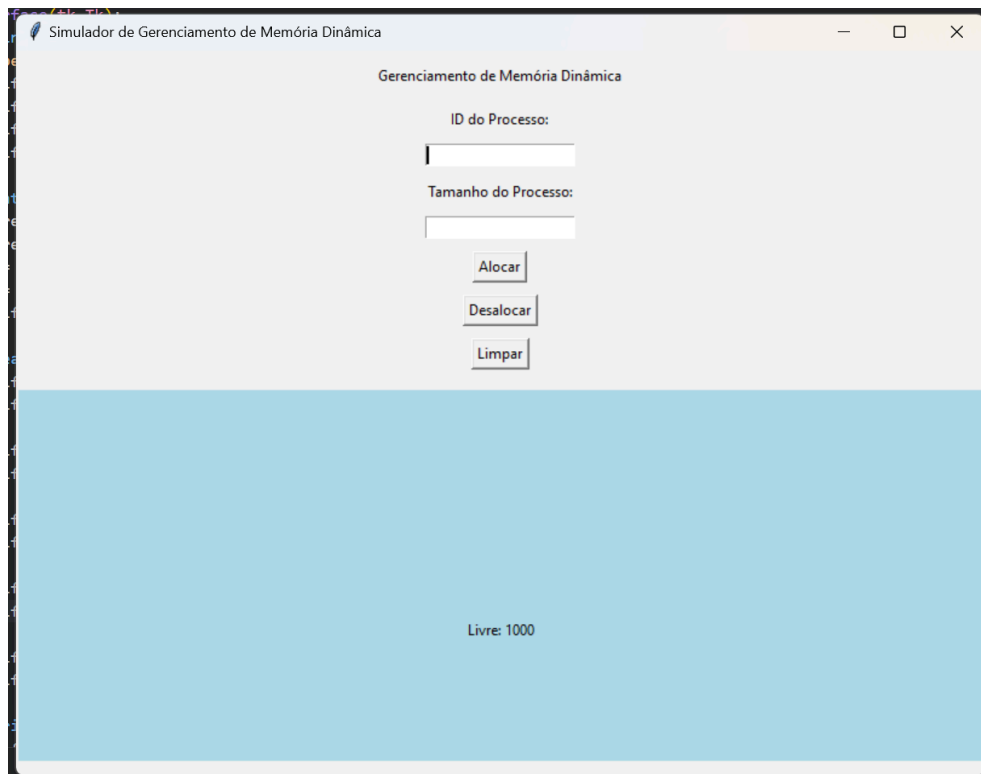
Além disso, também não é possível alocar um processo na memória, caso o tamanho do processo não tenha sido definido:



Como mencionado anteriormente, é possível limpar a memória, desalocando todos os seus processos de uma só vez. Após clicar no botão, há também a apresentação de uma mensagem de sucesso:

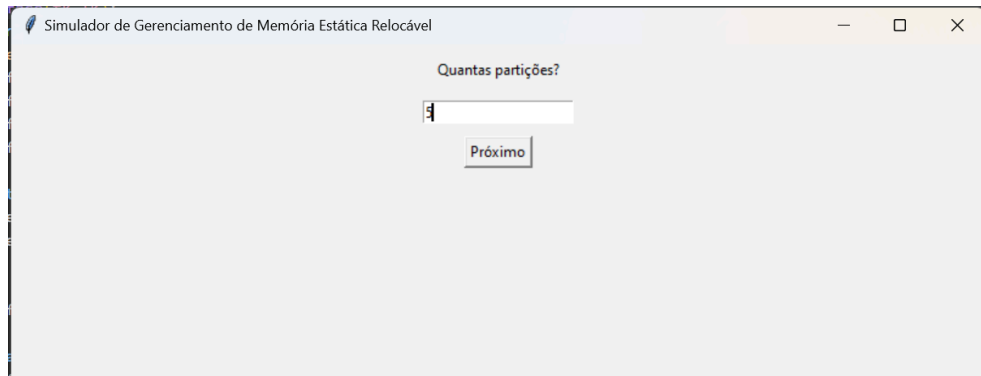


Seguindo, após a limpeza da memória, ela volta ao seu tamanho total estando livre:



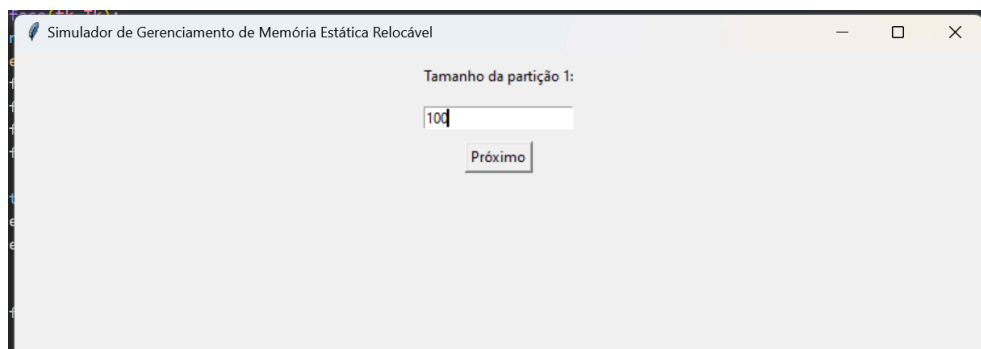
## Alocação Particionada Estática Relocável

Cada variação de gerenciamento de memória tem sua particularidade na hora da inicialização. No caso da alocação particionada estática relocável, o usuário precisa primeiramente definir em quantas partições quer separar a memória:

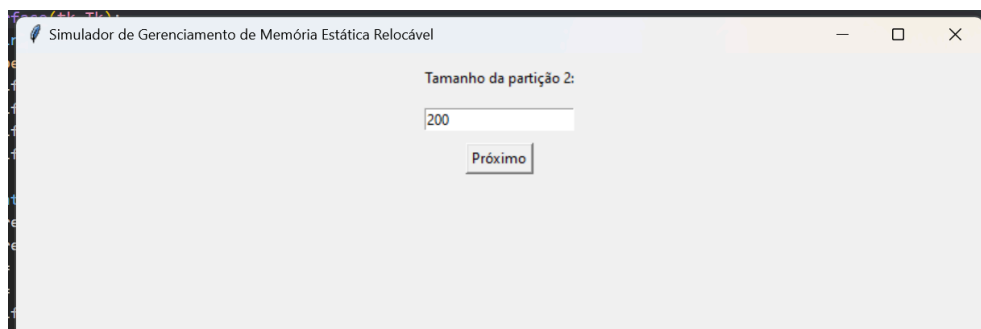


The screenshot shows a window titled "Simulador de Gerenciamento de Memória Estática Relocável". Inside, there is a text label "Quantas partições?". Below it is a text input field containing the number "3". To the right of the input field is a button labeled "Próximo".

Depois disso, o sistema irá questionar qual o tamanho de cada uma das partições:

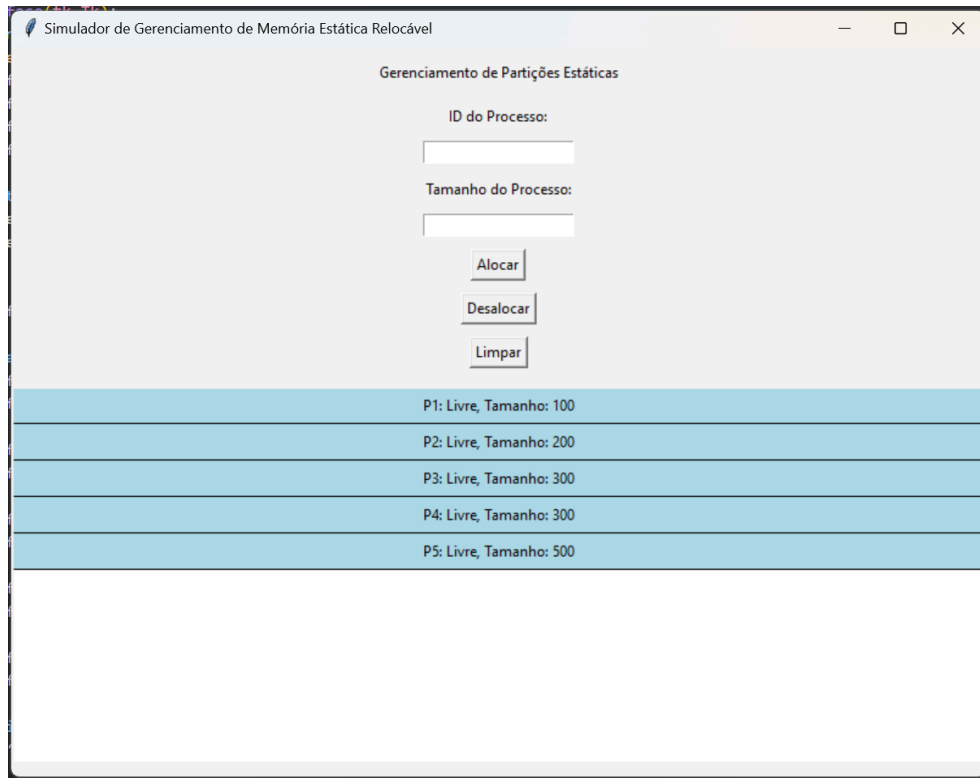


The screenshot shows a window titled "Simulador de Gerenciamento de Memória Estática Relocável". Inside, there is a text label "Tamanho da partição 1:". Below it is a text input field containing the number "100". To the right of the input field is a button labeled "Próximo".

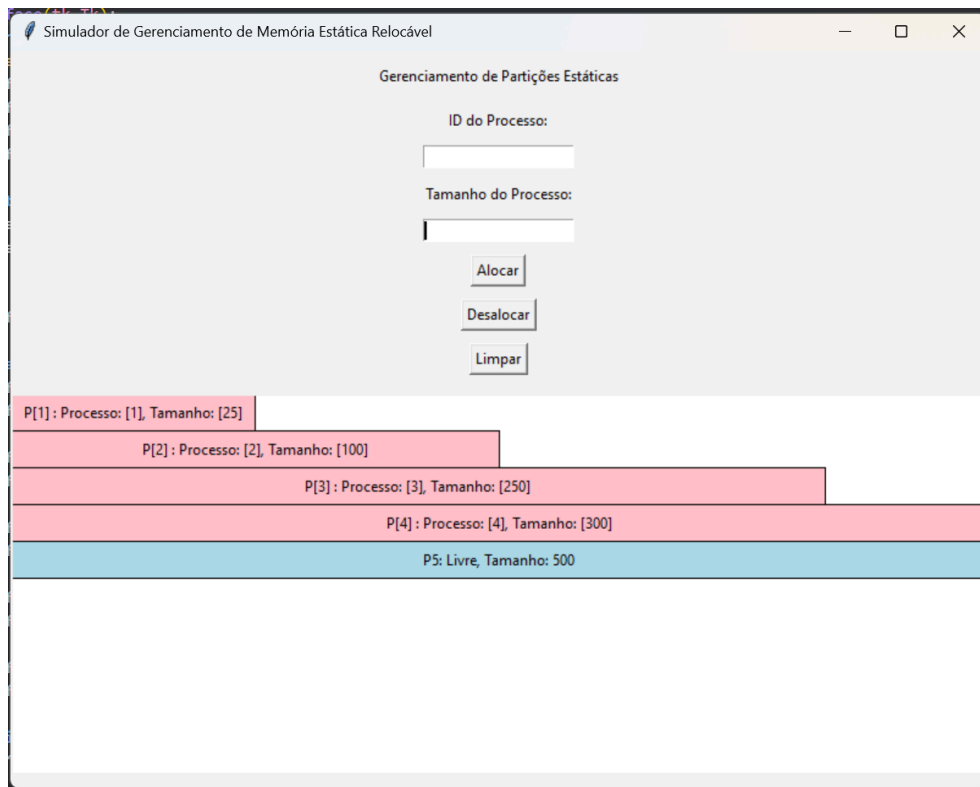


The screenshot shows a window titled "Simulador de Gerenciamento de Memória Estática Relocável". Inside, there is a text label "Tamanho da partição 2:". Below it is a text input field containing the number "200". To the right of the input field is a button labeled "Próximo".

Após definir a quantidade e o tamanho das partições, tem-se:



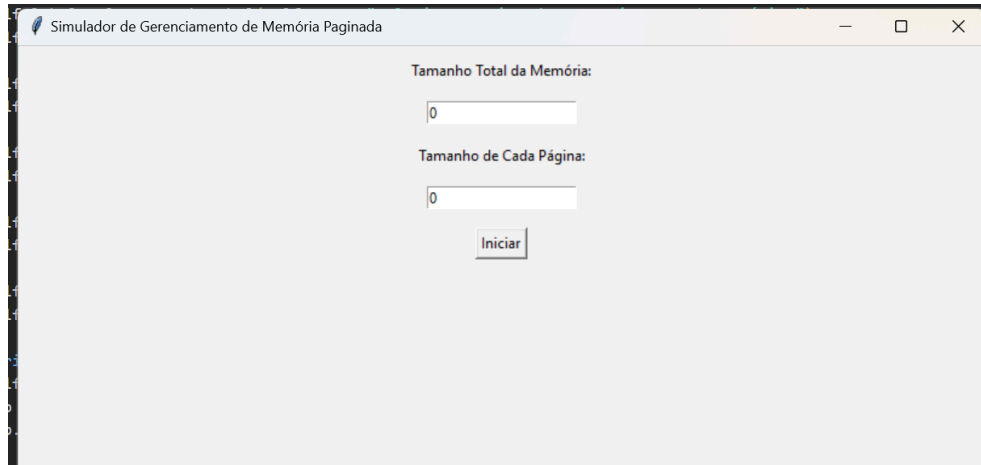
Como uma particularidade na interface gráfica dessa variação de gerenciamento, temos que cada uma das partições mostra proporcionalmente o seu grau de ocupação, como pode ser visto abaixo:





## Paginação

Na simulação de paginação, o usuário deve primeiramente definir o tamanho total da memória e o tamanho de cada página:



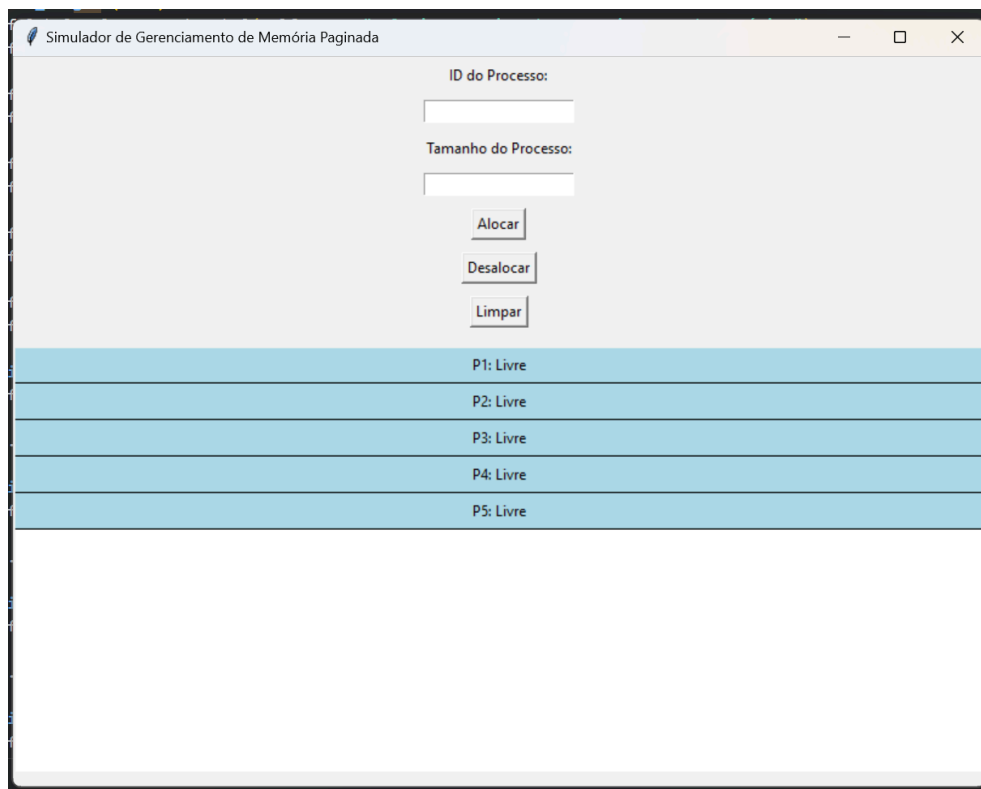
Simulador de Gerenciamento de Memória Paginada

Tamanho Total da Memória:

Tamanho de Cada Página:

Iniciar

Para demonstrar, vamos utilizar um tamanho total de 1000 e 200 para cada página:



Simulador de Gerenciamento de Memória Paginada

ID do Processo:

Tamanho do Processo:

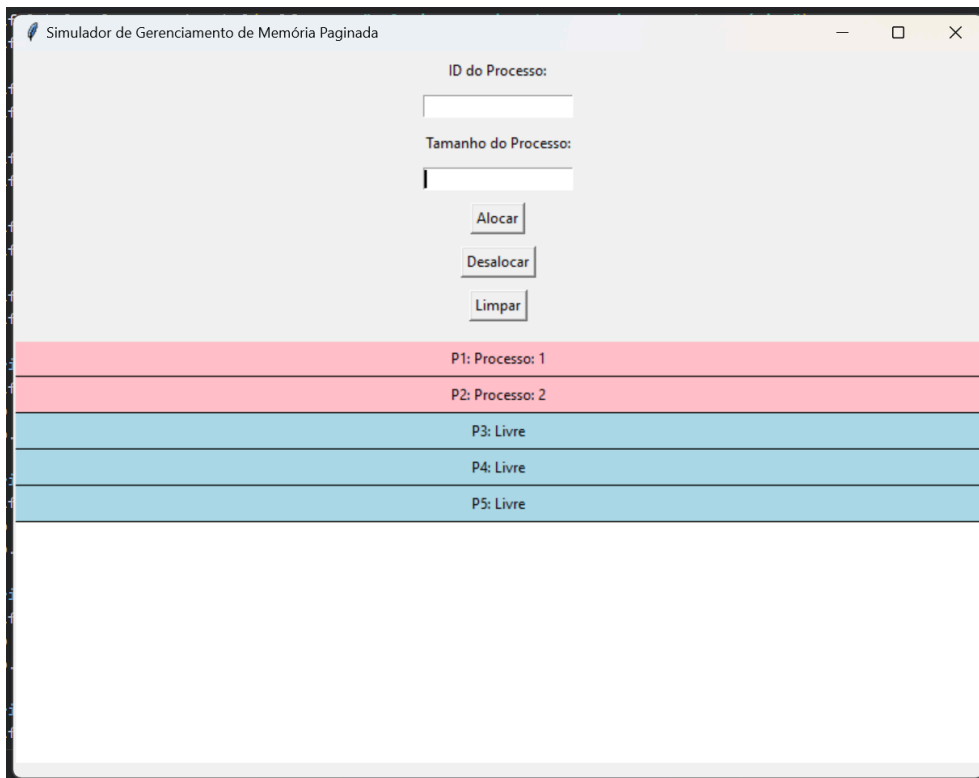
Alocar

Desalocar

Limpar

P1: Livre
P2: Livre
P3: Livre
P4: Livre
P5: Livre

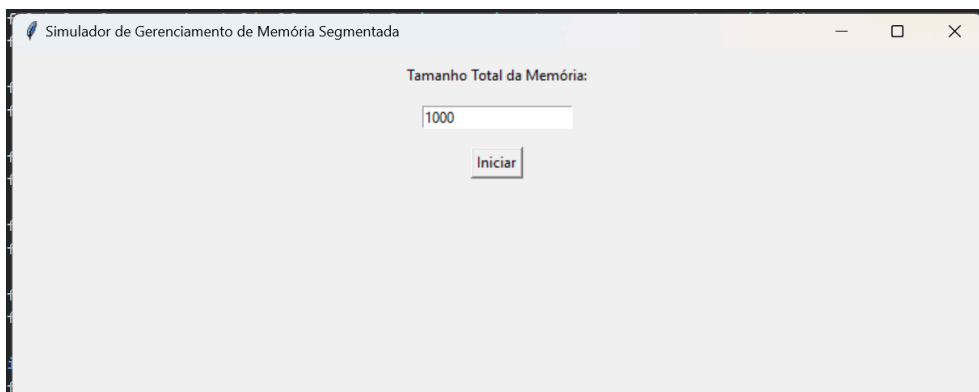
Após alocar alguns processos em memória, tem-se:



P1: Processo: 1
P2: Processo: 2
P3: Livre
P4: Livre
P5: Livre

## Segmentação

Para a variação de segmentação, novamente o usuário deve definir o valor total para a memória, nesse caso vamos utilizar 1000.



Após alocar alguns processos em memória, a tela apresentada mostra não o tamanho daquele segmento em si, mas sim o intervalo da memória ocupado pelo segmento do processo em questão.

Simulador de Gerenciamento de Memória Segmentada

ID do Processo:

Tamanho do Processo:

Alocar

Desalocar

Limpar

Segmento 1-200: Processo 1

Segmento 201-300: Processo 2

Segmento 301-1003: Livre