PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ RACIOCÍNIO ALGORÍTMICO PROF. JEAN PAUL BARDDAL

ATIVIDADE PRÁTICA 03 - SIMULAÇÃO DE ALOCAÇÃO DE MEMÓRIA

Você e sua equipe foram incumbidos de desenvolver um simulador de memória. A memória deverá ser simulada como uma matriz, onde cada posição determina se o espaço em questão está ocupado ou não, isto é, usando valores booleanos. O simulador a ser criado deve possuir as funcionalidades a seguir. Estas funcionalidades devem ser implementadas utilizando funções.

- **1.** Alocação da matriz de memória para simulação: O usuário deverá informar o número de linhas e colunas para a alocação inicial da memória. Deve-se garantir que o número de linhas e colunas sejam maiores que zero.
- **2. Visualização de memória:** O sistema deverá apresentar, de forma gráfica, a ocupação da memória. Um exemplo de ocupação de uma matriz de memória de tamanho 5x20 é apresentado abaixo, onde as regiões apresentadas como um 'X' estão ocupadas.

x		x		х	x	х	x	x		х		x	x			x		
	x	x	x		$ \mathbf{x} $		x			х	x		x	x		x	x	x
	x	x	x		x				x	x		x	x			x	x	x
X	x	x	x	x		x	x	x	x				x		x	x	x	×
			Х			Х	х	x			x	x		x	x	x		

- **3.** Alocação *first fit*: Esta estratégia de alocação encontra o primeiro espaço em memória cujo tamanho seja igual ou maior que o desejado e então realiza a alocação. No exemplo acima, caso uma alocação de tamanho 3 seja requisitada, ela seria realizada nas posições (0, 17), (0,18) e (0,19).
- **4. Alocação** *best fit*: Esta estratégia busca um espaço disponível que seja o mais adequado para o tamanho requisitado. Isto é, caso uma alocação de tamanho 'n' seja requisitada, um espaço de tamanho 'n' será buscado. Caso ele exista, a alocação será realizada. Caso contrário, um espaço de tamanho (n+1) será buscado e assim por diante, até que um espaço seja encontrado ou que seja determinado que nenhum espaço esteja disponível. No exemplo acima, caso uma alocação de tamanho 3 seja requisitada, ela deverá ocorrer nas posições (2, 6), (2,7) e (2,8).
- **5. Alocação worst fit:** O algoritmo de alocação worst fit realiza a alocação de memória na região com maior espaço livre contanto que ela seja suficientemente grande para comportar o tamanho requisitado. No exemplo acima, caso uma alocação de tamanho 2 seja requisitada, ela ocorrerá nas posições (0,17), (0,18), (0,19) e (1,0).
- **6. Desalocação:** O usuário deverá informar as coordenadas de início e fim para realizar a desalocação de memória. Deve-se garantir que todas as coordenadas informadas sejam válidas antes de realizar a desalocação.
- **7. Modo de teste:** O modo de teste deve gerar uma matriz de tamanho 5000x5000 e efetuar 1 milhão de alocações aleatórias usando as estratégias *first fit, best fit* e *worst fit*. Ao final, deve-se apresentar as seguintes estatísticas: **número de alocações bem-sucedidas**, malsucedidas, e o tempo gasto durante todo o teste. Importante: as alocações (tamanhos) a serem testadas em cada abordagem de alocação devem ser iguais.

NOTAS IMPORTANTES

- A memória deve, obrigatoriamente, ser implementada como uma estrutura bidimensional.
- As linhas de memória devem ser tratadas como uma região contígua.
- Esta atividade deverá ser realizada em TRIOS.
- Entregas atrasadas terão dedução de 0,5 (meio ponto) por dia de atraso.
- Todos os integrantes da equipe deverão realizar o teste de auditoria. A não participação resultará no decréscimo de 30% do valor total da atividade.
- O código-fonte deve ser entregue em arquivos Python convencionais, isto é, arquivos com extensão .py.
- A data de entrega e da vista dos trabalhos estão dispostas no ambiente virtual de aprendizagem (*Blackboard*).