

INSTITUTO FEDERAL DO TRIÂNGULO MINEIRO

Alunos: David Marques, Íthalo Silva, Paulo Henrique e Larissa Santos

Disciplina: Sistemas Operacionais

Data: 05/10/2023

RELATÓRIO DO TRABALHO (Colaboração do Grupo)

O grupo conseguiu organizar e dividir as tarefas em partes do código para implementação completa do mesmo, utilizando o CodeBlocks como IDE para o desenvolvimento do código, usando a linguagem C++, e também recorremos a uma estrutura de lista para simular o escalonamento de Processos SRT (Shortest Remaining Time);

Estabelecemos horários para tirar Dúvidas a respeito do código, e criamos um grupo para esse contexto;

DESENVOLVIMENTO DO CÓDIGO

Como foi explicado na aula o SRT é um algoritmo de escalonamento de CPU que seleciona o processo com o menor tempo restante para executar, priorizando assim os processos de menor duração. O Nosso código é composto por várias estruturas e funções que permitem ao usuário inserir processos e simular a execução deles usando o SRT. Abaixo, vou explicar as principais partes do código e como ele funciona:

ESTRUTURAS DE DADOS

1. struct No: Esta estrutura define um nó que contém informações sobre um processo, incluindo seu tempo de processamento (TempProc), seu nome (NomeProc) e um ponteiro para o próximo nó (prox) na lista.
2. struct Lista: Esta estrutura define uma lista encadeada que será usada para armazenar os processos. Ela possui um ponteiro para o início (cabeca) e para o final (cauda) da lista.

FUNÇÕES DA ESTRUTURA LISTA

- Lista(): O construtor da estrutura "Lista" inicializa a lista vazia.
- bool vazia(): Esta função verifica se a lista está vazia.
- void inserirFinal(int TempProcesso, char NomeProcesso[]): Esta função insere um novo processo no final da lista. Ela cria um novo nó, preenche os dados do processo e ajusta os ponteiros da lista adequadamente.
- No* encontrarMenorProcesso(): Esta função percorre a lista em busca do processo com o menor tempo restante e retorna um ponteiro para esse processo.
- void executarSrt(): Essa função simula a execução dos processos usando o algoritmo SRT. Ela continua até que todos os processos tenham sido concluídos, imprimindo informações sobre a execução de cada processo.
- void removerProcesso(No* processo): Esta função remove um processo da lista, dado um ponteiro para ele.
- void imprimir(): Esta função imprime o estado atual da lista de processos.

FUNÇÃO MAIN

A função main é a função principal do programa. Ela começa pedindo ao usuário para inserir o número de processos a serem simulados (TempP) e verifica se esse número está dentro do intervalo permitido (de 1 a 15).

Em seguida, ela entra em um loop para solicitar ao usuário o nome e o tempo de processamento de cada processo. Cada processo é inserido na lista usando a função inserirFinal.

Depois que todos os processos são inseridos, a lista é impressa com a função imprimir, e a função executarSrt é chamada para simular a execução dos processos de acordo com o algoritmo SRT.

CONCLUSÃO

Este código em C implementa um simulador simples de escalonamento de processos usando o algoritmo Shortest Remaining Time (SRT). Ele permite ao usuário criar uma lista de processos, especificando seus nomes e tempos de processamento, e simula a execução desses processos de acordo com o SRT. Este código é uma demonstração útil de como os algoritmos de escalonamento de processos podem ser implementados em sistemas operacionais.