

Relatório SO

João Verdilheiro, nº38156

Julho 2021

1 Introdução

O trabalho realizado implementa um sistema de 5 estados, com sistema de gestão de memória Best-Fit. No seu desenvolvimento foi utilizado uma implementação de queue, [1], sendo que foi alterada para que fosse adequada ao trabalho em questão. O código está dividido em algumas partes para que seja mais fácil analisar, sendo por exemplo métodos relativamente a instruções, memória, escalonamento ou a própria implementação de Queues. O trabalho foi realizado com a possibilidade de introduzir um ficheiro txt para receber o input das funções, sendo que basta alterar o pathfile no método input para que o documento seja escolhido.

2 Gestão de memória, Best-Fit

A memória utiliza uma matriz de 200x2 para guardar a que processo pertence a informação contida em cada espaço. O sistema de gestão de memória recorre a alguns métodos para conseguir calcular a memória necessária para o processo, não só a memória para guardar as instruções necessárias como também para as suas variáveis. Depois de saber a memória necessária para armazenar o processo em questão irá analisar se é ou não possível que este seja guardado, escolhendo o espaço de memória que este irá "encaixar" melhor.

3 Transição entre estados, Round Robin

A transição entre os diferentes estados dos processos é feito com o auxílio das Queues, sendo que existem duas filas para cada estado, Ready ou Block. Em cada instante será verificado se as filas não estão vazias e no caso se algum dos processos deve ser retirado da fila. No caso do estado Run, o primeiro da fila Ready, terá de ser verificada a instrução que deve ser corrida, sendo que o seu resultado poderá ter efeitos na fila ou não, por exemplo o caso da instrução HLT.

4 Instruções

As instruções são tratadas sempre que algum programa está numa instância em run. Sendo isto verdade será verificado o valor correspondente à sua instrução e serão feitas as devidas alterações. Existe um cuidado especial no caso das instruções de jump, verificando sempre se estas instruções estão dentro do espaço de armazenamento do processo em questão, avisado e terminado o processo sempre que tal não se verificar. A instrução Fork é também analisada com detalhe pois irá dar origem a um novo processo sendo que terá de ser verificado se é possível que este seja criado ou não.

Referências

- [1] Tutorials Point. Queue program in c. https://www.tutorialspoint.com/data_structures_algorithms/queue_program_in_c.htm. Accessed: 2021-07-11.