

## SEGUNDA LISTA DE EXERCÍCIOS DE ESTRUTURAS DE DADOS

1. O estacionamento Picoense contém uma única alameda que guarda até 10 carros. Existe uma entrada e uma saída, de tal forma que quando um determinado carro entra ele fica no final da fila e o primeiro que chegou sempre fica próximo a saída, ou seja, todas as vezes que um carro sai todos os outros devem ser manobrados um espaço para frente. Faça um programa em C, onde o usuário entrará com o número da placa do carro e 'E' se estiver entrando no estacionamento e 'S' se estiver saindo do estacionamento. O programa deve emitir uma mensagem sempre que um carro entrar ou sair do estacionamento. Quando um carro chegar, a mensagem deve especificar se existe ou não vaga no estacionamento, esse não tiver vaga, simplesmente o carro vai embora, não existe fila de espera. Quando um carro sair, a mensagem deverá incluir o número de vezes em que o carro foi manobrado para fora do estacionamento para permitir que outros carros saíssem, além de mostrar quantos carros foram manobrados para que ele saísse.

Obs.: Não se esqueça que todas as vezes que um carro vai sair os que estão na frente devem ser manobrados para o final da fila, mas ao final os carros que estava na frente do carro que saiu devem continuar no mesmo lugar de antes, ou seja, todos os demais carros devem ir para o fim da fila, mas não deve ser contabilizado como manobra.

2. Refaça o Exercício 1 substituindo fila estática por fila dinâmica.
3. Faça um programa em C onde o usuário digita uma expressão matemática no modo in-fixa e então o programa verifica se a expressão é válida, depois use pilha estática para converter para o modo pós-fixa.

Obs. 1: A expressão deve ser lida em uma string, e a string deve ser no máximo de 100 caracteres.

Obs. 2: A expressão só pode conter dígitos e os operadores: (, +, -, \*, /); cada número e operador deve ser separado por um espaço em branco, podendo a sim um número possuir mais de um dígito.  
Exemplo: 130 + 50 - 25

Obs. 4: A expressão deve sempre iniciar com operando ou abre parênteses e terminar com operando ou fecha parênteses.

4. Refaça o Exercício 3 substituindo pilha estática por pilha dinâmica.
5. Escalonador é uma parte do Sistema Operacional que escolhe de uma fila de prontos o próximo processo a ocupar o processador. Suponha que nosso escalonador mantenha 3 filas dinâmicas, de acordo com a prioridade do processo.

Cada processo possui as seguintes informações: o seu número, o tempo de processamento que o mesmo necessita em segundos e a prioridade do mesmo para que ele possa ser inserido na fila de acordo com a sua prioridade.

O escalonador deve sempre escolher primeiro os processos da fila de maior prioridade e somente passar para outra fila quando todos os processos da fila atual tiver sido executado pelo menos 2 vezes, ou seja, ficar vazia.

Quando o escalonador retira o primeiro processo da fila para usar o processador, o mesmo só pode ficar 1s utilizando o processador, terminado este tempo o escalonador diminui de 1 o tempo de processamento deste processo e verifica seu valor. Se o valor do tempo de processamento atual for 0, o processo é finalizado, ou seja, retirado da fila, caso contrário, será verificado quantas vezes o processo passou nesta fila para decidir em qual fila o mesmo será inserido. Se o processo passou somente 1 vez será colocado ao final da mesma fila, se passou 2 vezes será colocado ao final da próxima fila de prioridade abaixo da atual. Caso ele já esteja na última fila, ou seja, na fila de menor prioridade, o processo será inserido ao final desta fila até que o mesmo termine independente do número de vezes que já passou por ela.

Faça um programa em C que tenha um menu de opções para inserir processos em uma fila do escalonador, para o escalonador escolher o processo(caso não existia nenhum processo na fila emitir uma mensagem), para mostrar os processos das filas, para mostrar o próximo processo que irá utilizar o processador, mostrar quantos processos tem em cada fila, para mostrar quanto tempo falta para executar os processos de uma determinada fila e para mostrar quanto tempo de processamento ainda falta para chegar em um determinado processo e mostrar quanto tempo de processamento ainda restam para terminar cada fila e todas as filas.

**Equipe:** os programas podem ser feitos em dupla, mas os relatórios são individuais. Se os programas forem feitos em dupla, a dupla deve ser identificada no envio do código.

**Data de Entrega:** data primeira prova escrita

**Entregar:** Código Fonte, Relatório(Conforme Modelo em PDF)

**Forma de Entrega:** pelo SIGAA, caso tenha algum problema enviar por e-mail (julianaoc@gmail.com).

**Entrevista Individual:** agendar horário com a Professora.

**Processo da Entrevista:** para cada aluno será sorteado 2 programas, dentre os entregues pelo aluno, para que o mesmo explique os 2 programas.