

## **Projeto 2 – Estruturas de Dados I SCC-0202**

**Data de entrega: 24/10/2019**

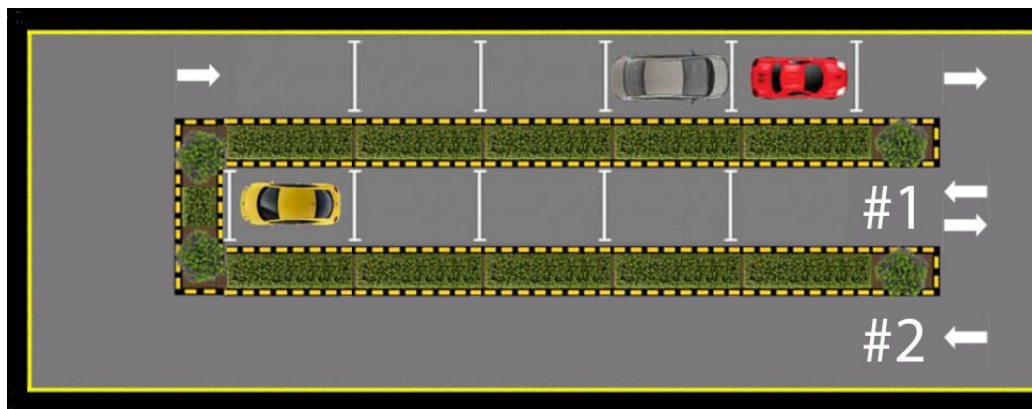
### **Observações**

- Este projeto deverá ser realizado em duplas;
- Somente uma submissão por dupla é permitida;
- Especificar em um arquivo de texto junto com o código-fonte os nome e NUSP de cada um da dupla;
- O código-fonte deve ser comprimido em um único arquivo, no formato *zip*, com o nome “Projeto2.zip”, e entregue via Atividades do TIDIA;
- O projeto deve ser desenvolvido usando somente a linguagem de programação C;
- As duas partes do projeto devem estar no mesmo arquivo;
- Será utilizada uma ferramenta para verificar plágio nos códigos, sendo penalizados com nota 0 aqueles trabalhos que tenham plágio confirmado; Portanto, não serão permitidas cópias em hipótese alguma;
- O projeto será avaliado em termos de corretude e legibilidade do código;
- Em caso de dúvidas, envie email para [eugenio.cabral@usp.br](mailto:eugenio.cabral@usp.br)

# Parte 1 - Estacionamento PnF

Nota: 7 pontos

O Estacionamento PnF é um estacionamento que possui dois pátios para guardar os carros, com capacidade para 15 carros, sendo 5 carros no Pátio 1 e 10 carros no Pátio 2. Em ambos os pátios a passagem é muito estreita, assim os carros são sempre estacionados um atrás do outro. A diferença entre os locais encontra-se na saída dos veículos. No Pátio 1, tanto a entrada quando a saída de um veículo são realizadas pelo mesmo local, enquanto no Pátio 2 a entrada é realizada por um local e a saída pelo seu lado vertical oposto. A Figura a seguir ilustra a configuração dos pátios do estacionamento.



Atualmente, há um problema recorrente no estacionamento, os carros frequentemente bloqueiam a saída de outros carros. Por exemplo, há 2 carros estacionados no Pátio 2, e após algum tempo o dono do segundo carro estacionado (cor cinza) volta para retirar o seu carro, como o carro do cliente não é o carro mais próximo da saída, todos os carros que estão bloqueando seu caminho são retirados do estacionamento (neste caso, o carro vermelho) a fim do carro do cliente ser manobrado para fora do pátio. O mesmo acontece no Pátio 1, com frequência o dono de um carro que não é o mais próximo da saída chega para retirar o seu carro e o mesmo está bloqueado pelos carros à sua frente. Então, uma solução para este problema foi organizar os carros de modo que o próximo carro a sair esteja sempre com a sua passagem livre. Portanto, no pátio 2 quando um carro sai do estacionamento os carros que ainda estacionados no pátio 2 são movidos uma vaga a frente, ou seja, na Figura, assim o que carro vermelho sai do estacionamento, a sua vaga é preenchida pelo carro de cor cinza que, por sua vez, libera a sua vaga para um novo veículo. Já no pátio 1, a movimentação dos carros não é necessária pois os carros que estão prestes a sair estão estacionados na última vaga disponível. Assim, a fim de automatizar o fluxo das saídas, o seu objetivo função é desenvolver um programa que decida em qual pátio o novo carro deve ser estacionado. Note que o Pátio 1 e 2, possuem um comportamento similar as estruturas pilha e fila, respectivamente. Estas estruturas deve ser implementadas com base em uma lista encadeada respeitando as regras do Estacionamento PnF apresentadas abaixo.

## Regras do Estacionamento PnF

O valor do estacionamento é de 3 reais por hora e seu funcionamento é baseado nos novos carros que chegam para estacionar. A cada carro, é realizado o *check-in* para registrar algumas informações do carro, em seguida, é realizado o *check-out* dos carros estacionados com base no horário de chegada do novo carro, e por fim, é verificado se há *disponibilidade* em algum dos pátios para o novo carro. Caso seja possível, o carro deve ser alocado no pátio adequado, caso contrário o carro não poderá estacionar. Os detalhes sobre cada um dos passos são apresentados a seguir.

(1) **Check-in:** Ao chegar no estacionamento o dono do veículo deve informar a placa do carro (como um inteiro de 4 dígitos), o horário de chegada, bem como o total de horas o qual o carro ficará estacionado. Tanto o horário de entrada quanto o horário de saída devem ser armazenados em um inteiro, assumindo somente hora cheia no padrão 24h, ou seja, após as 12h deve-se armazenar o valor 13 em vez de 1. O tempo de permanência do carro será sempre em horas cheias, não são toleradas horas fracionadas (minutos). O horário de entrada do novo carro sempre será maior que o horário de entrada do último carro. Por fim, não são aceitos carros com placas duplicadas. Caso a placa de um carro ainda estacionado seja informada no *registro*, o sistema deve notificar que o carro já está estacionado no pátio.

(2) **Check-out:** Antes de estacionar um novo carro, é necessário imprimir na tela os carros (placa do veículo e valor a ser pago - com desconto ou não, vide parte 2) o qual o seu horário de saída esteja expirado e removê-los (*check-out*) dos pátios 1 e 2. Para isto, deve se comparar o horário de saída dos carros já estacionados com o horário de chegada do novo carro. Assim, deve-se remover todos os carros de ambos os pátios os quais o seu horário de saída seja menor ou igual ao horário de chegada do novo carro.

(3) **Disponibilidade:** Se ambos os pátios estiverem vazios, o novo carro deve ser estacionado no Pátio 1. Caso, exista carro no Pátio 1 e nenhum carro no Pátio 2, o carro deve ser estacionado no Pátio 2 e vice-versa. Caso exista pelo menos 1 carro em cada um dos pátios e pelo menos um dos pátios encontram-se com vagas disponíveis (não esteja lotado), as seguintes regras devem ser adotadas: No Pátio 1, um novo carro só pode ser estacionado se, o seu horário de saída for menor ou igual ao o horário de saída do último carro estacionado nele (topo). No Pátio 2, um novo carro só pode ser estacionado se seu horário de saída for maior ou igual ao horário de saída do último carro estacionado nele (calda). Caso o novo carro satisfaça as condições de ambos pátios, o novo carro poderá ser estacionado em qualquer um dos pátios.

(4) **Rejeição do Carro:** Um carro será rejeitado se o estacionamento estiver cheio, ou o seu horário de entrada ou saída não estiverem dentro do horário de funcionamento do estacionamento (8h às 22h), ou nenhuma das regras de *Disponibilidade* não foram satisfeitas. Uma mensagem deve ser emitida apresentando a justificativa da rejeição do carro.

## Funcionalidades do Programa

Para este programa, é esperado uma TAD do carro contendo as seguintes informações: placa do carro (inteiro), horário de chegada (inteiro), horário de saída (inteiro) e um campo desconto (float) - vide Parte 2. O programa deve apresentar um menu principal com 3 (três) funções: (1) Registrar Carro, (2) Imprimir Carros, (3) Sair. Os detalhes sobre cada uma das funcionalidades são apresentados a seguir.

- **Registrar Carro:** Deve-se implementar sequencialmente as regras de *Check-in*, *Check-out*, *Disponibilidade* e *Rejeição do Carro*.
- **Imprimir Carros:** Deve imprimir na tela todos os carros que estão estacionados atualmente nos pátios. Deve ser exibido a placa do veículo, em qual pátio ele está estacionado (1 ou 2) e o valor total a ser pago (considerar desconto caso aplicado - vide Parte 2).
- **Sair:** O programa deve ser imediatamente finalizado, e todas as estruturas devem ser desalocadas da memória.

## Parte 2 - Sorteio de Descontos

Nota: 3 pontos

Para fidelizar os clientes do estacionamento, os proprietários decidiram introduzir uma regra de *Sorteio de Descontos*, onde o carro sorteado recebe um desconto de 10% sob o valor total a pagar. Os carros podem ser sorteados mais de uma vez, porém os descontos não são cumulativos. Esta regra deve ser introduzida após a regra de **Check-out** e antes da regra de **Disponibilidade**.

O sorteio só é realizado caso o estacionamento esteja com pelo menos 25% da sua capacidade total ocupada (considerando ambos os pátios) e quando o horário de entrada do novo carro coincide com os horários de sorteio pré-estabelecidos (9h, 12h, 15h, e 18h). Entretanto, o sorteio só deve considerar os carros atualmente estacionados e não o novo carro.

A lógica por trás do sorteio é baseada em uma *roleta de descontos*, onde cada posição da roleta representa um carro. Ao rodar a roleta, a posição em que o cursor estiver apontando será o carro que receberá o desconto. A roleta pode ser abstraída em um vetor contendo os carros atualmente estacionados (de ambos dos pátios), o qual não importa a ordem. O cursor da roleta inicialmente está na posição zero do vetor. Então, um número aleatório deve ser gerado, o qual indicará quantas posições o cursor deverá percorrer para determinar qual carro será sorteado. Caso o número aleatório seja maior que o tamanho do vetor, então o cursor deve voltar para a sua posição inicial até o que número de posições percorridas seja igual ao número aleatório gerado. Esta é a mesma estratégia usada em uma Array Circular. Uma mensagem deve ser emitida exibindo a placa do carro que foi sorteado e o valor total (com desconto) que o cliente deverá pagar.