

# Enunciado do Projeto 1 - IAED 2023/24

## Data de entrega

01 de abril 2024, às 19h59

## LOG alterações

8mar24 - Publicação do enunciado.

14mar24 - Exemplo de facturação adicionado

## 1. Introdução

Pretende-se a construção de um sistema de gestão de parques de estacionamento. O seu sistema deverá permitir a definição de parques e o registo de entradas e saídas de veículos, assim como a sua consulta e facturação.

A interacção com o programa deverá ocorrer através de um conjunto de linhas compostas por uma letra (comando) e um número de argumentos dependente do comando a executar. Pode assumir que todo o input fornecido respeitará os tipos indicados, por exemplo onde é esperado um valor inteiro decimal nunca será introduzida uma letra. Os possíveis comandos são listados na tabela seguinte e indicam as operações a executar.

### Comando Acção

q termina o programa

p Cria um parque de estacionamento com o regime de facturação ou lista os parques existentes

e Regista a entrada de um veículo

s Regista a saída de um veículo

v Lista as entradas e saídas de um veículo

f Mostra a facturação de um parque de estacionamento

r Remove um parque de estacionamento do sistema

## 2. Especificação do problema

O objectivo do projeto é ter um sistema de gestão de parques de estacionamento até um máximo de 20 parques de estacionamento.

Cada parque é caracterizado por um nome, uma capacidade máxima e um regime de facturação. No nome que descreve o parque podem ocorrer caracteres brancos (espaços ou tabulador horizontal \t). Neste caso, o nome é representado entre aspas. Caso não contenha caracteres brancos, o nome pode ser delimitado por aspas ou não. O nome nunca contém o carácter aspa ou o carácter \n na sua descrição.

O valor a facturar é definido em intervalos de 15 minutos. Dependendo da duração da permanência do veículo no parque, o valor a facturar em cada período varia. O regime de facturação de todos os parques é definido por três valores:

X : o valor por cada 15 minutos na 1ª hora;

Y : o valor por cada 15 minutos após a 1ª hora;

Z : o valor máximo diário (24 horas);

Nos primeiros 4 períodos de 15 minutos é cobrado X por cada período. A partir da 1ª hora é cobrado Y por cada período adicional de 15 minutos. No entanto, se o período de permanência no parque for inferior a 24 horas, então o valor máximo a cobrar não pode ser superior a Z. Note-se que no tarifário temos sempre que  $Z > Y > X$ .

Os veículos podem permanecer num parque por mais de 24 horas. Nesse caso, é aplicado o valor máximo diário Z a cada período completo de 24 horas que permanecer no parque. O valor a cobrar pelo período remanescente é calculado de acordo com o definido para um período inferior a 24 horas como descrito no parágrafo anterior.

### 3. Dados de Entrada

O programa deverá ler os dados de entrada a partir da linha de comandos do terminal. Nenhuma linha de comandos excede BUFSIZ bytes (8192 bytes na maioria dos sistemas). Durante a execução do programa as instruções devem ser lidas do terminal (standard input) na forma de um conjunto de linhas iniciadas por um carácter, que se passa a designar por comando, seguido de um número de informações dependente do comando a executar; o comando e cada uma das informações são separados por pelo menos um carácter branco.

### 4. Compilação e teste

O compilador a utilizar é o gcc com as seguintes opções de compilação: -O3 -Wall -Wextra -Werror -Wno-unused-result. Para compilar o programa deve executar o seguinte comando:

```
$ gcc -O3 -Wall -Wextra -Werror -Wno-unused-result -o proj1 *.c
```

O programa deverá escrever no standard output as respostas aos comandos apresentados no standard input. As respostas são igualmente linhas de texto formatadas conforme definido anteriormente neste enunciado. Tenha em atenção ao número de espaços entre elementos do seu output, assim como a ausência de espaços no final de cada linha. Procure respeitar escrupulosamente as indicações dadas.

### 5. Entrega do Projeto

Será criado um repositório git para cada aluno desenvolver e submeter o projeto. Este repositório será criado no GitLab da RNL e será activado quando da publicação deste enunciado. Na sua submissão do projeto deve considerar os seguinte pontos:

1. Os ficheiros de desenvolvimento do projeto (.c e .h) devem estar na raiz do repositório.
2. Apenas a última versão no repositório da RNL será considerada para avaliação.
3. Actualizações no repositório recomeçam um período de espera de 10 minutos antes da avaliação

automática ser executada.

Data limite de entrega: 01 de abril de 2024, às 19h59m.

Bónus: Entregas até 27 de março de 2024, às 19h59, terão bónus adicional, salvo alterações após essa data.

## **6. Avaliação do Projeto**

Na avaliação do projeto serão consideradas as seguintes componentes:

1. Desempenho da funcionalidade do programa realizado (0 a 16 valores).
2. Qualidade do código entregue, como comentários, estruturação, e modularidade (entre -4 e +4 valores).

## **7. Dicas para Desenvolvimento do Projeto**

Desenvolvam o projeto de forma incremental, testando localmente cada comando antes de actualizarem o repositório remoto.

Passos sugeridos:

- Corrija o código de forma incremental, garantindo que compila sem erros nem warnings.
- Garanta que lê e escreve o input/output correctamente, respeitando os formatos especificados.
- Desenvolva e teste cada comando individualmente.