

Instituto Politécnico de Viseu

Escola Superior de Tecnologia e Gestão de Viseu

Departamento de Informática

Unidade Curricular: Algoritmos e Programação

Relatório Relativo ao Trabalho Prático

Tema: Parque de Estacionamento

Realizado por: Guilherme Félix – 25172

João Cruz – 25178

Rodrigo Pereira – 27450

Viseu, 2024

Instituto Politécnico de Viseu

Escola Superior de Tecnologia e Gestão de Viseu

Departamento de Informática

Relatório relativo ao Trabalho Prático

Curso de Licenciatura em Engenharia Informática

Unidade Curricular de Algoritmos e Programação

Parque de Estacionamento

Ano Letivo 2023/24

Viseu, 2024

**ÍNDICE**

[**1. Estruturas e acesso a ficheiros 3**](#_6tcqub4llmen)

[1.1. Estruturas 3](#_6qcyvhy60gk2)

[1.2. Ler entradas 4](#_98ck11ltp08)

[1.3. Carregar Clientes 4](#_8ecudayqk9p6)

[1.4. Ler Entradas e Saídas 5](#_c1sr9ips8uao)

[***2. Navegação por menus 6***](#_30j0zll)

[**2.1. ExibirMenuClientes() 7**](#_1fob9te)

[**2.2. ExibirMovimentoVeiculos() 7**](#_1t3h5sf)

[**3. Funções do MainMenu() 8**](#_2s8eyo1)

[3.1. Adicionar Cliente 8](#_uywnj9nmptmy)

[3.2. Retirar Cliente 9](#_glx1baglcq7e)

[3.3. Alterar Dados de ClientePara esta função é perguntado ao utilizador qual o id de cliente a alterar, e serão perguntadas as mesmas questões da função para adicionar cliente. 9](#_icpt3ylqugn5)

[3.4. Mostrar todos os Clientes 10](#_8jo0ec5922c9)

[3.5. Atualizar Clientes 10](#_e74grd1c65qj)

[3.6. Data Final da Subscrição 11](#_8omtd35h5opv)

[3.7. Registar Entrada 11](#_a9n0fr1gzvet)

[3.8. Registar Saída 12](#_y65g9zx5ciz6)

[3.9. Mostrar Entradas 12](#_9yjnos2sqekf)

[3.10. Mostrar Saídas 13](#_g8uqoboc6w5g)

[**3.11. Informação de ocupação 13**](#_3rdcrjn)

[**3.12. Mapa de ocupação 14**](#_26in1rg)

[**3.13. Saídas e valores pagos 15**](#_lnxbz9)

[3.14. Listar clientes com subscrição 17](#_4q0lfcc1bujm)

[***4. Conclusões 18***](#_z337ya)

Introdução

Este projeto em C aborda a gestão de um parque de estacionamento de um supermercado, composto por três andares, cada um com 20 linhas e 25 colunas, totalizando 1500 lugares. O programa registra entradas e saídas de veículos por meio de matrículas, calculando o custo de estacionamento na saída. Além disso, o sistema gera um mapa que indica os lugares ocupados e a quantidade total de lugares disponíveis. A implementação eficiente em linguagem C visa garantir um funcionamento direto e ágil, abordando de forma prática as necessidades de gestão de estacionamento.

O trabalho explora detalhes da lógica do sistema, destacando a alocação de lugares, o registo de movimentação e o cálculo de custos, com reflexões sobre possíveis melhorias para aprimorar a eficiência global.

No capítulo 2, haverá uma explicação geral na organização do código, através da navegação de menus.

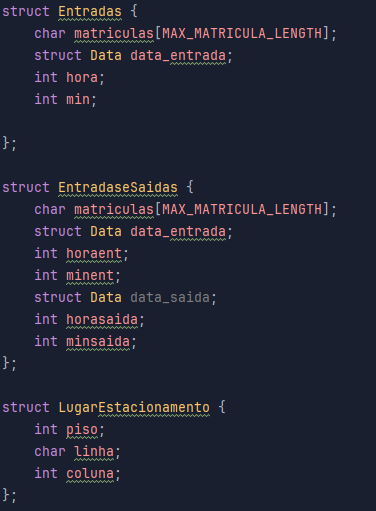
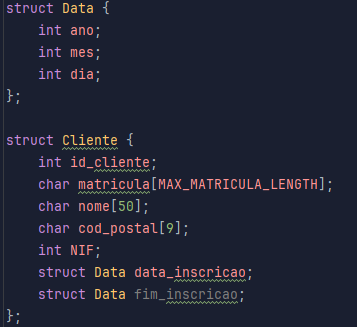
No seguinte, vamos explicar o que cada opção no menu principal faz, seguido de exemplos.

# Estruturas e acesso a ficheiros

Para este trabalho recorremos a diversas estruturas para separar variáveis relevantes.

Através de 3 funções, abrimos ficheiros distintos, retiramos dados de cada coluna e associamos aos mesmos a uma variável na devida estrutura

## Estruturas



Na estrutura “Data” é registado ano, mês e dia, sendo esta usada na data de subscrição de um cliente, entrada e saída de veículos.

Na estrutura “Cliente”, são registadas informações relevantes em relação ao cliente, tais como, Nome, NIF, cod\_postal, e matrícula do carro associado ao mesmo.

Na estrutura “Entradas” é registado as matrículas dos carros que entraram no parque, e é usada a estrutura Data para registar o ano, mês, dia e o seu devido horário de entrada.

Na estrutura “EntradaseSaidas” é registado as matrículas dos carros que entraram e saíram do parque, e é usada a estrutura Data para registar o ano, mês, dia e o seu devido horário de entrada e saída.

Na estrutura “LugarEstacionamento” é registado o piso, a linha e a coluna de um lugar ocupado por um veículo.

## Ler entradas

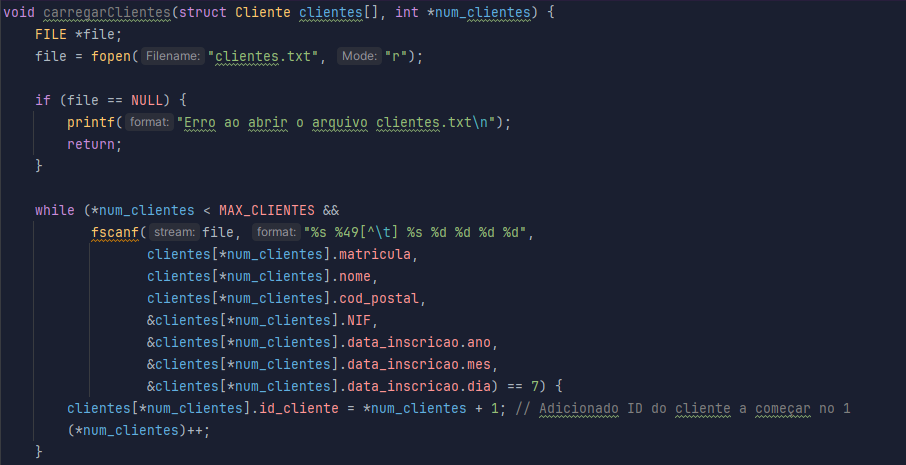
Para esta função, abrimos o ficheiro “entradas.txt” e registamos o dado relevante de cada coluna, nomeadamente, a matrícula, data de entrada e o lugar no qual o veículos ficou estacionado, sendo este lugar catalogado por piso, linha e coluna.



Cada coluna lhe será associada uma variável,e os dados serão guardados em memória para alteração, e leitura em iterações futuras.

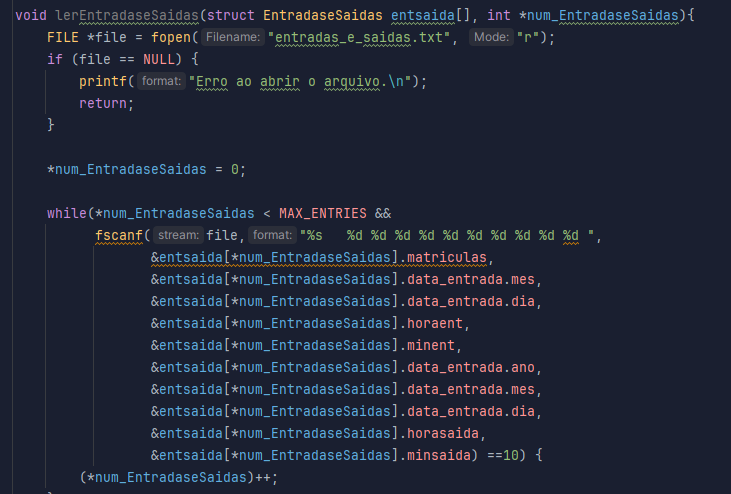
## Carregar Clientes

Para a função “carregarClientes” ocorre o mesmo processo, é aberto o ficheiro “clientes.txt”, que serve para mostrar os clientes que possuem subscrição no parque.



## Ler Entradas e Saídas

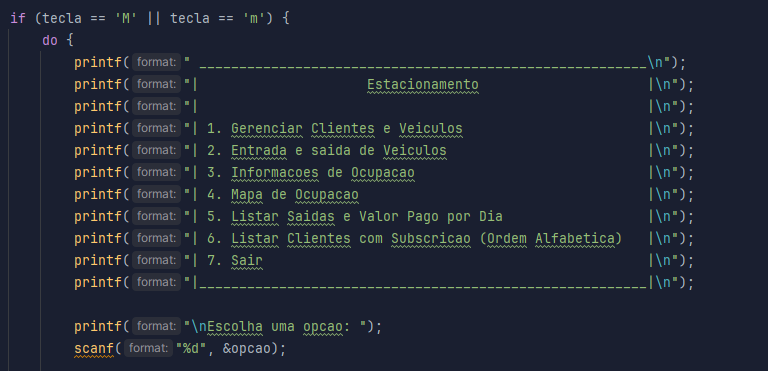
Uma vez mais, a função lerEntradasSaidasa abre o ficheiro entradas\_e\_saidas.txt, e, associa às estruturas a matrícula, e a sua respectivas data de entrada e saída do estacionamento.



# Navegação por menus

Algumas das principais funcionalidades incluem a gestão de clientes e veículos, registo de pagamentos de subscrições, entrada e saída de veículos, exibição de informações de ocupação, visualização de um mapa de ocupação, listagem de saídas e valores pagos por dia, listagem de clientes com subscrição e a opção de sair do programa.

Através do main(), o utilizador pode executar cada uma destas tarefas através das opções no menu, sendo ExibirMenuClientes() e ExibirMovimentoVeiculos() submenus para tarefas mais específicas.

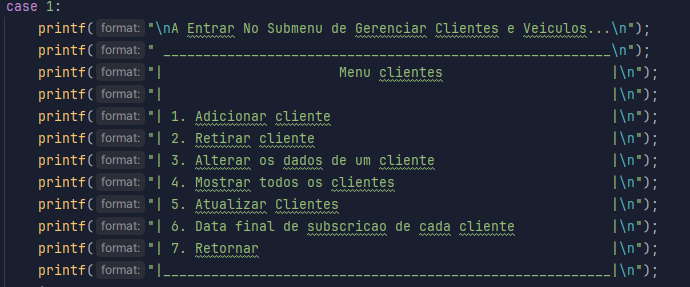


Através do switch case, acedemos aos submenus e às suas devidas funções.

No case 1 acede-se ao submenu “Menu Clientes”.  
No case 2 acede-se ao submenu “Menu Veículos”

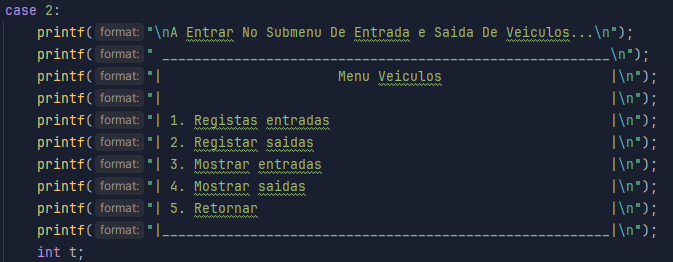
## ExibirMenuClientes()

Este menu permite ao utilizar adicionar, remover, alterar um cliente com subscrição, respetivamente, do ficheiro.

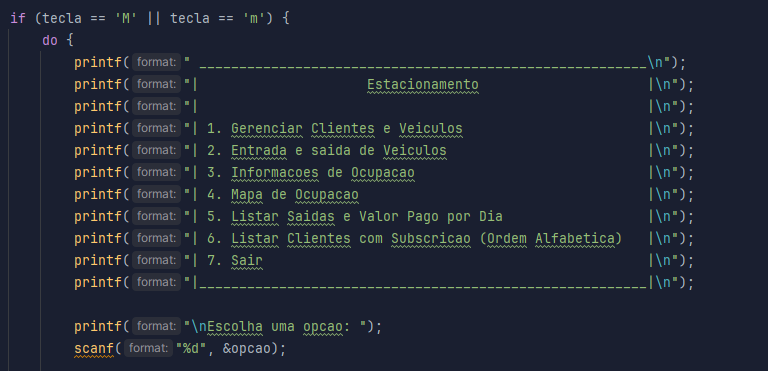


## ExibirMovimentoVeiculos()

Neste menu, o utilizador pode registar a entrada e saída de veículos.



# Funções do MainMenu()



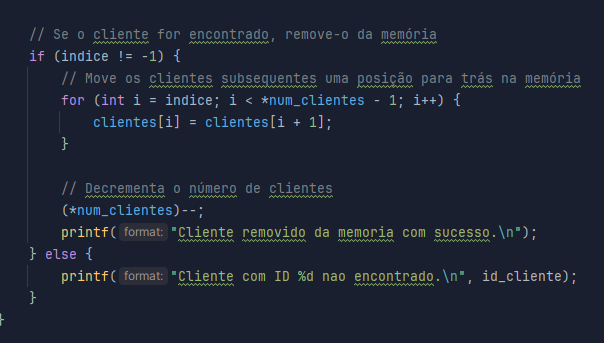
## Adicionar Cliente

Nesta função, é questionado ao utilizador os dados para um cliente que efetuou uma subscrição, nomeadamente, a matrícula a ele associado, NIF, código postal, e a data no qual efetuou subscrição.



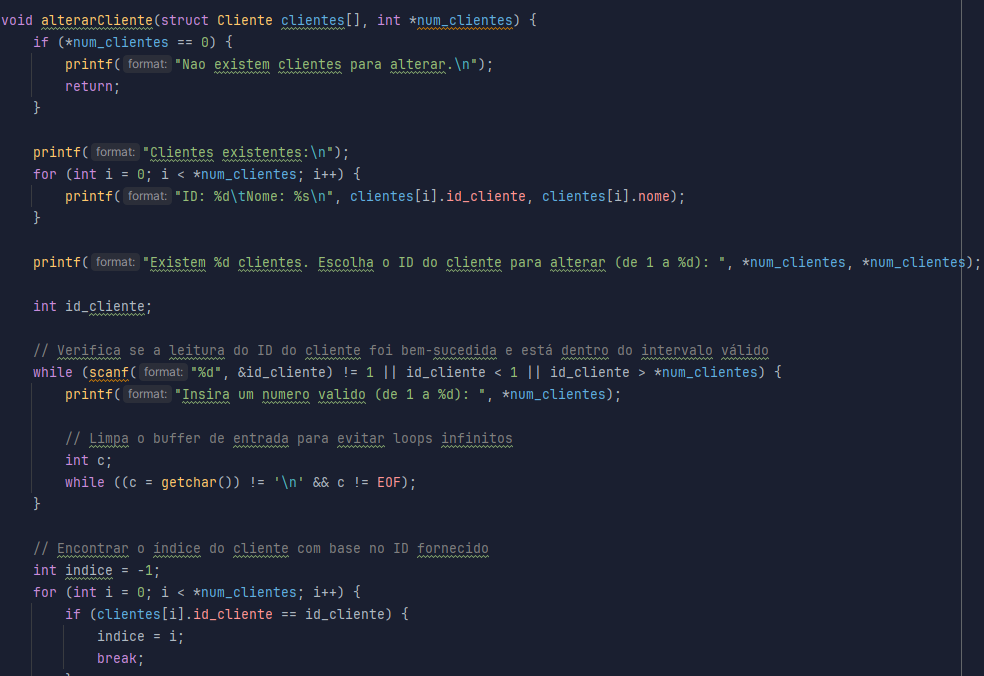
## Retirar Cliente

Nesta função, mostramos ao utilizador quais os clientes já existentes e pedimos qual o id do cliente a remover, de seguida apaga o conteúdo de toda a linha da memória.

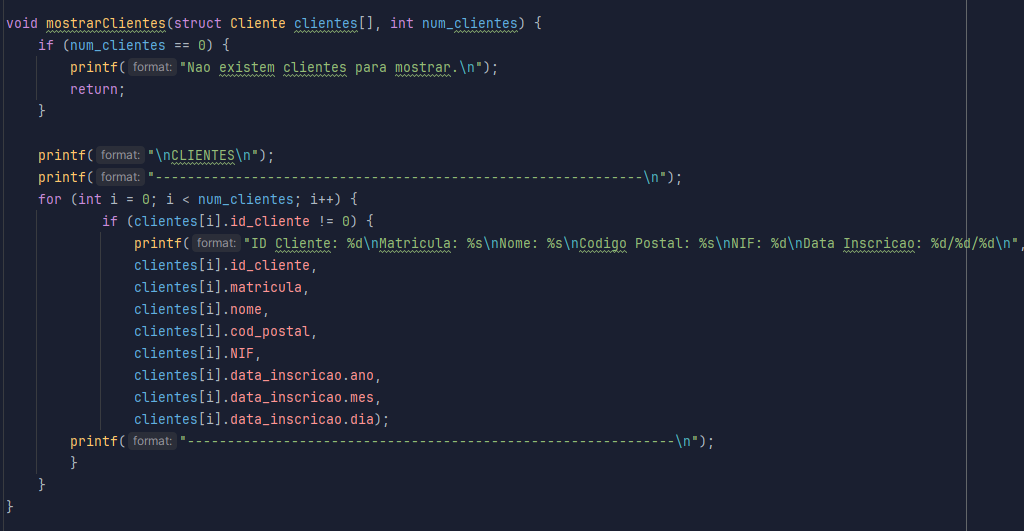


## Alterar Dados de Cliente

Para esta função é perguntado ao utilizador qual o id de cliente a alterar, e serão perguntadas as mesmas questões da função para adicionar cliente.

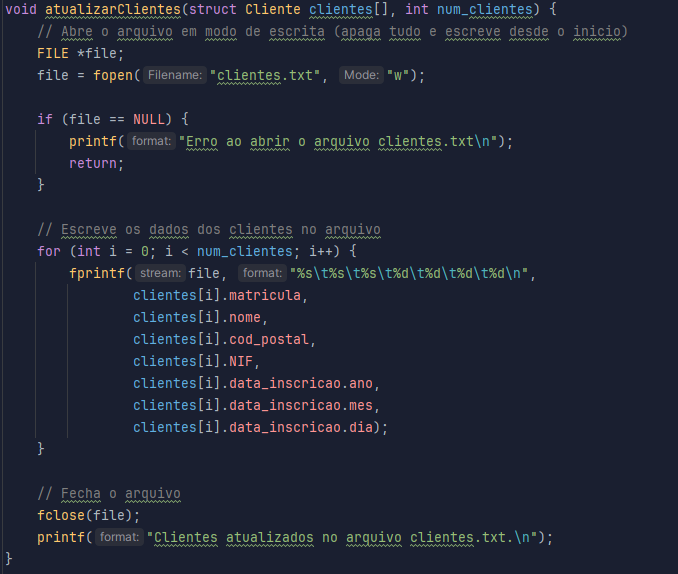


## Mostrar todos os Clientes

Para esta função, o programa percorre linha por linha e da print ao conteúdo da mesma, sendo que, estes são os dados do cliente.

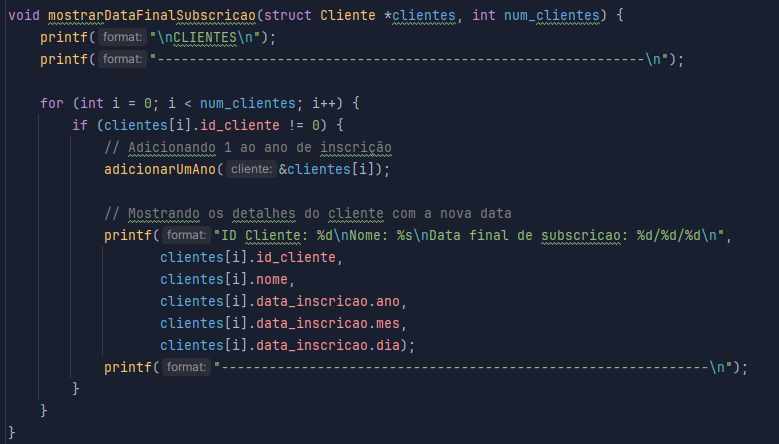
## Atualizar Clientes

Esta função, depois das alterações feitas pelo utilizador, irá abrir o ficheiro clientes.txt e fazer as alterações pedidas pelo utilizador.



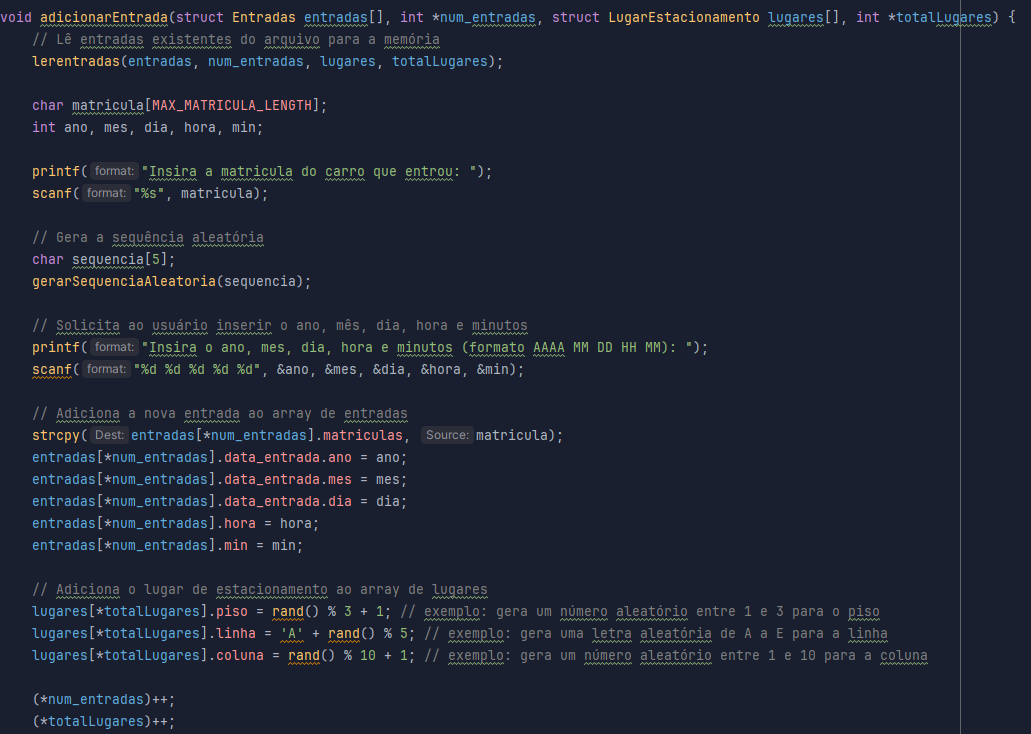
## Data Final da Subscrição

Para esta função, para cada cliente com subscrição, mostra o seu nome, e a data de quando a mesma acaba.



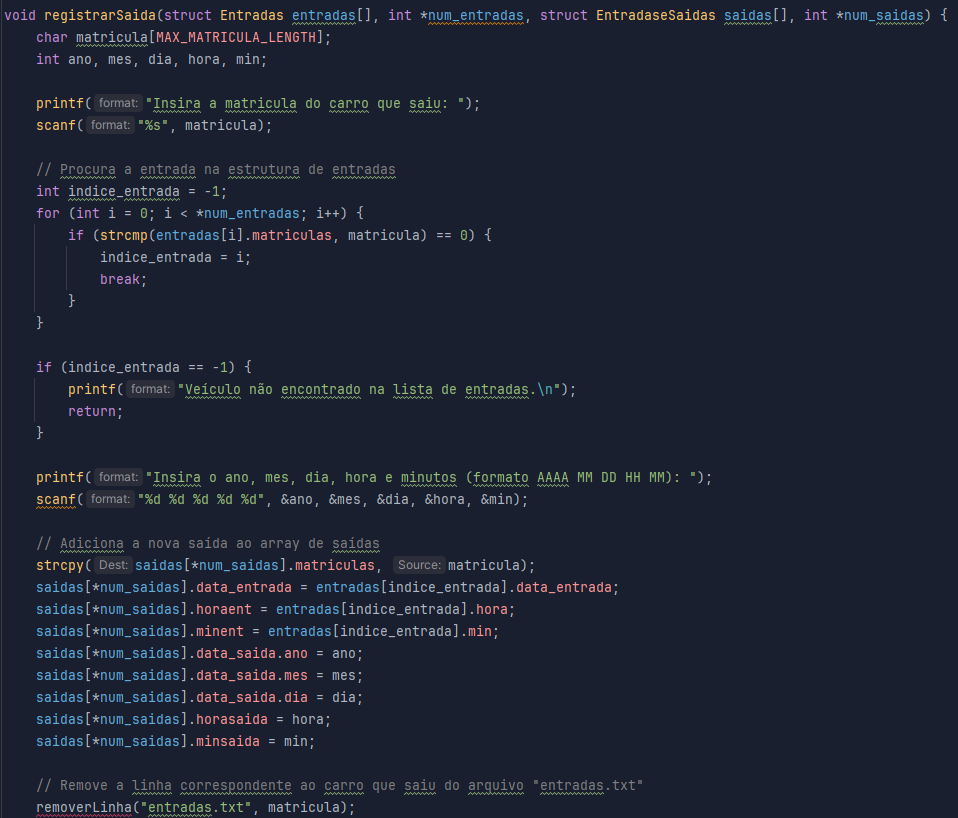
## Registar Entrada

Para esta função, é pedido qual a matrícula, ano, mês, dia, hora e minuto de entrada, e após isso o programa fornece automaticamente um lugar livre no parque, escolhido aleatoriamente:



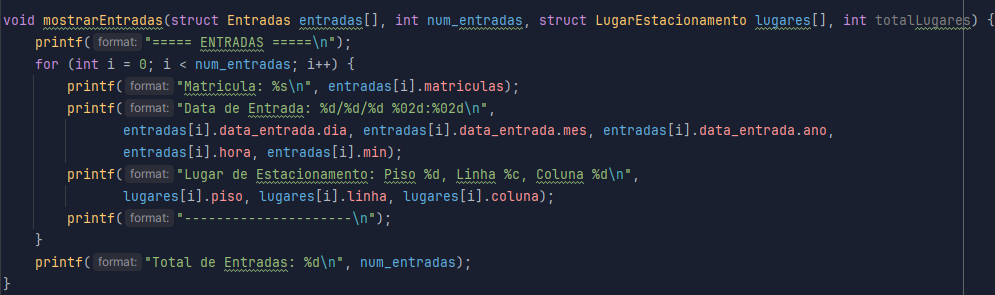
## Registar Saída

Nesta função, registamos um veículos que entrou no parque através da matrícula, anotamos a data de entrada e saída e escrevemos no ficheiro “entradas.txt”.



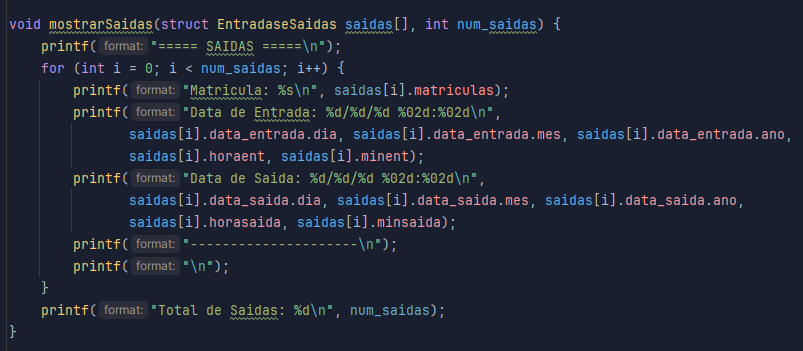
## Mostrar Entradas

Esta função, mostra a lista de entrada de veículos, nomeadamente a sua matrícula data de entrada, e o lugar de estacionamento a ele atribuído.



## Mostrar Saídas

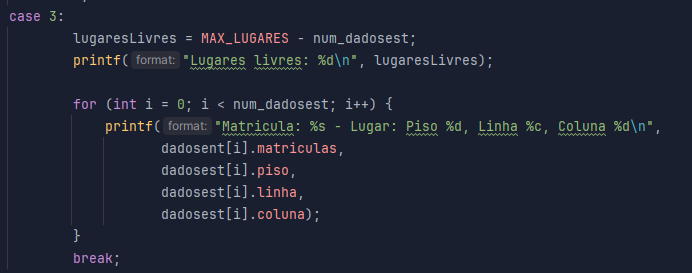
Nesta função, mostra a lista de saída de veículos, nomeadamente a sua matrícula e data de saída.



## Informação de ocupação

A terceira opção do menu, “Informações de ocupação” analisa os dados, conta os lugares ocupados, subtrai ao total, e mostra o número de lugares livres ao utilizador.

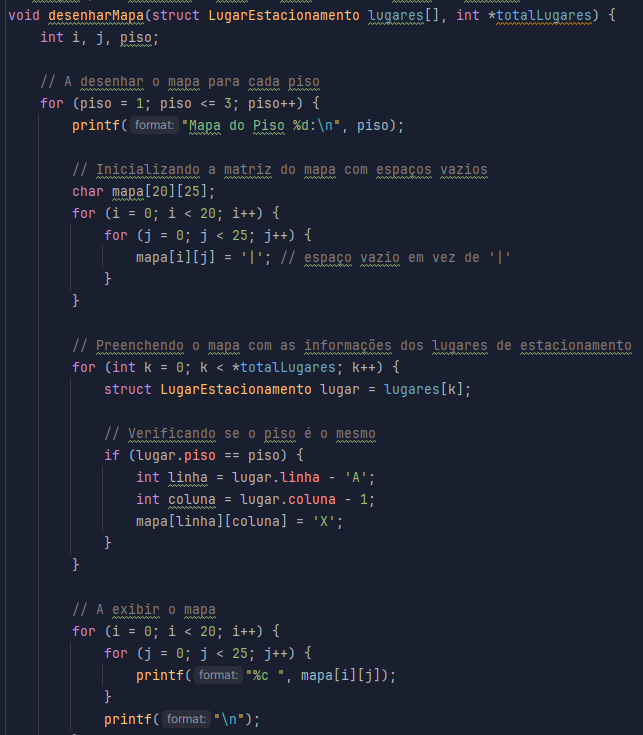
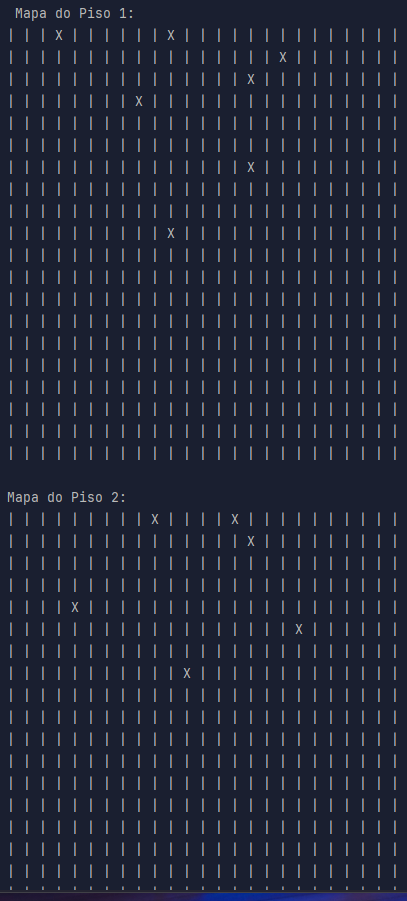
Não só isso mas também, mostra o piso, linha e coluna do lugar onde uma matrícula especificada se encontra.



## Mapa de ocupação

A quarta opção do menu, através da função “lerentradas”, retira qual piso, linha e coluna se encontra estacionado, através do código do lugar.

Por fim, desenha 3 grelhas 20x25, preenchendo com um “X” os lugares ocupados.



## Saídas e valores pagos

Na quinta opção, através das informações nos ficheiros de entrada e saída, é feito um cálculo de quanto o cliente tem de pagar, seguindo os padrões da tabela de preços:

Por fim, mostra os pagamentos ao utilizador através da consola, e cria um ficheiro de texto com a lista de todos os pagamentos efetuados.



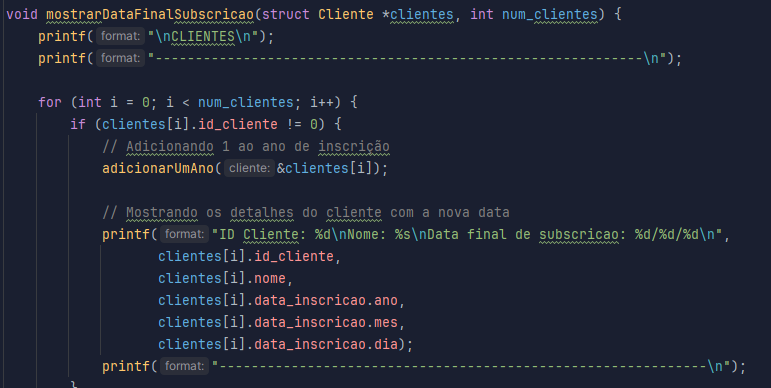






## Listar clientes com subscrição

Para a última, o programa acede aos dados registados pela função carregar clientes, e escreve por ordem alfabética os clientes presentes no ficheiro, já que este se destina apenas a clientes com subscrição.



# Conclusões

O projeto de gestão de estacionamento em linguagem C proporcionou uma implementação prática e modular. A estrutura organizada do código facilita a manutenção e expansão futura.

Apesar de fornecer funcionalidades abrangentes, como o registo de entradas/saídas e a exibição de informações de ocupação, algumas áreas, como o registo de pagamentos de subscrições, exigem desenvolvimento adicional.

A aplicação demonstrou eficiência na utilização de estruturas de decisão e leitura de dados de arquivos. No entanto, otimizações adicionais podem ser exploradas para aprimorar a eficiência global.