

## Relatorio

Generated by Doxygen 1.9.3



<b>1 Data Structure Index</b>	<b>1</b>
1.1 Data Structures	1
<b>2 File Index</b>	<b>3</b>
2.1 File List	3
<b>3 Data Structure Documentation</b>	<b>5</b>
3.1 Esquema Struct Reference	5
3.1.1 Detailed Description	5
3.2 Meio Struct Reference	5
3.2.1 Detailed Description	6
3.3 Pedido Struct Reference	6
3.3.1 Detailed Description	6
3.4 Pedidos Struct Reference	6
3.4.1 Detailed Description	6
<b>4 File Documentation</b>	<b>7</b>
4.1 Estruturas.h File Reference	7
4.1.1 Detailed Description	7
4.2 Estruturas.h	8
4.3 Funcoes.c File Reference	8
4.3.1 Detailed Description	10
4.3.2 Function Documentation	10
4.3.2.1 buscaMeio()	10
4.3.2.2 criarMeio()	11
4.3.2.3 criarPedido()	11
4.3.2.4 custoPedido()	11
4.3.2.5 distribuirMeios()	12
4.3.2.6 existeMeio()	12
4.3.2.7 existePedido()	12
4.3.2.8 guardarMeios()	13
4.3.2.9 guardarPedidos()	13
4.3.2.10 insertMeio()	13
4.3.2.11 insertPedido()	14
4.3.2.12 interfacePrincipal()	14
4.3.2.13 lerMeios()	14
4.3.2.14 lerPedidos()	15
4.3.2.15 listarPlanoMeio()	15
4.3.2.16 menuMeio()	15
4.3.2.17 menuPedido()	16
4.3.2.18 menuPrincipal()	16
4.3.2.19 mostraMeio()	16
4.3.2.20 mostraMeios()	17

4.3.2.21 mostraPedido()	17
4.3.2.22 mostraPedidos()	17
4.3.2.23 procuraVeiculoSuficiente()	17
4.3.2.24 removerMeio()	18
4.3.2.25 removerPedido()	18
4.3.2.26 verificaAutonomia()	19
4.4 Funcoes.h File Reference	19
4.4.1 Detailed Description	20
4.4.2 Function Documentation	21
4.4.2.1 criarMeio()	21
4.4.2.2 criarPedido()	21
4.4.2.3 custoPedido()	21
4.4.2.4 distribuirMeios()	22
4.4.2.5 existeMeio()	22
4.4.2.6 existePedido()	23
4.4.2.7 guardarMeios()	23
4.4.2.8 guardarPedidos()	23
4.4.2.9 insertMeio()	24
4.4.2.10 insertPedido()	24
4.4.2.11 interfacePrincipal()	25
4.4.2.12 lerMeios()	25
4.4.2.13 lerPedidos()	25
4.4.2.14 listarPlanoMeio()	25
4.4.2.15 menuMeio()	26
4.4.2.16 menuPedido()	26
4.4.2.17 menuPrincipal()	26
4.4.2.18 mostraMeio()	28
4.4.2.19 mostraMeios()	28
4.4.2.20 mostraPedido()	28
4.4.2.21 mostraPedidos()	29
4.4.2.22 procuraVeiculoSuficiente()	29
4.4.2.23 removerMeio()	29
4.4.2.24 removerPedido()	30
4.4.2.25 verificaAutonomia()	30
4.5 Funcoes.h	30
4.6 main.c File Reference	31
4.6.1 Detailed Description	31
4.6.2 Function Documentation	32
4.6.2.1 main()	32

# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">Esquema</a>	Estrutura do conjunto de meios de mobilidade elétrica e sua respetiva quantidade . . . . .	5
<a href="#">Meio</a>	Estrutura do meio de mobilidade elétrica . . . . .	5
<a href="#">Pedido</a>	Struct que representa um pedido realizado por um utilizador . . . . .	6
<a href="#">Pedidos</a>	Struct para um conjunto de pedidos e sua respetiva quantidade . . . . .	6



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">Estruturas.h</a>	Nesta biblioteca terá todas as structs do programa . . . . .	7
<a href="#">Funcoes.c</a>	Neste ficheiro .c será introduzido todas as funções do programa exceto a main . . . . .	8
<a href="#">Funcoes.h</a>	Nesta biblioteca será introduzido código referente a todas as assinaturas de funções . . . . .	19
<a href="#">main.c</a>	Neste ficheiro .c apenas terá a função main para execução do programa . . . . .	31





## Chapter 3

# Data Structure Documentation

### 3.1 Esquema Struct Reference

Estrutura do conjunto de meios de mobilidade elétrica e sua respetiva quantidade.

```
#include <Estruturas.h>
```

#### Data Fields

- int **qt\_meios**
- [Meio](#) **meio** [TAM\_A]

#### 3.1.1 Detailed Description

Estrutura do conjunto de meios de mobilidade elétrica e sua respetiva quantidade.

The documentation for this struct was generated from the following file:

- [Estruturas.h](#)

### 3.2 Meio Struct Reference

Estrutura do meio de mobilidade elétrica.

```
#include <Estruturas.h>
```

#### Data Fields

- char **cod** [TAM\_T]
- char **tipo** [TAM\_T]
- float **custo**
- int **autonomia**
- int **id**

### 3.2.1 Detailed Description

Estrutura do meio de mobilidade elétrica.

The documentation for this struct was generated from the following file:

- [Estruturas.h](#)

## 3.3 Pedido Struct Reference

Struct que representa um pedido realizado por um utilizador.

```
#include <Estruturas.h>
```

### Data Fields

- int **num**
- int **nif**
- int **tempo**
- int **distancia**
- char **cod** [TAM\_T]

### 3.3.1 Detailed Description

Struct que representa um pedido realizado por um utilizador.

The documentation for this struct was generated from the following file:

- [Estruturas.h](#)

## 3.4 Pedidos Struct Reference

Struct para um conjunto de pedidos e sua respetiva quantidade.

```
#include <Estruturas.h>
```

### Data Fields

- int **qt\_pedidos**
- [Pedido](#) **pedido** [TAM\_A]

### 3.4.1 Detailed Description

Struct para um conjunto de pedidos e sua respetiva quantidade.

The documentation for this struct was generated from the following file:

- [Estruturas.h](#)

## Chapter 4

# File Documentation

### 4.1 Estruturas.h File Reference

Nesta biblioteca terá todas as structs do programa.

#### Data Structures

- struct [Pedido](#)  
*Struct que representa um pedido realizado por um utilizador.*
- struct [Pedidos](#)  
*Struct para um conjunto de pedidos e sua respetiva quantidade.*
- struct [Meio](#)  
*Estrutura do meio de mobilidade elétrica.*
- struct [Esquema](#)  
*Estrutura do conjunto de meios de mobilidade elétrica e sua respetiva quantidade.*

#### Macros

- #define **TAM\_T** 10
- #define **TAM\_A** 100

#### 4.1.1 Detailed Description

Nesta biblioteca terá todas as structs do programa.

##### Author

João Apresentação ( [a21152@alunos.ipca.pt](mailto:a21152@alunos.ipca.pt))  
Gonçalo Cunha ( [a21145@alunos.ipca.pt](mailto:a21145@alunos.ipca.pt))  
Carlos Barbosa ( [a21157@alunos.ipca.pt](mailto:a21157@alunos.ipca.pt))

##### Version

0.1

##### Date

2022-01-22

##### Copyright

Copyright (c) 2022

## 4.2 Estruturas.h

[Go to the documentation of this file.](#)

```

1
14 #ifndef ESTRUTURAS
15 #define ESTRUTURAS
16 #define TAM_T 10 //tamanho para texto
17 #define TAM_A 100 //tamanho para structs
22 typedef struct
23 {
24     int num, nif, tempo, distancia;
25     char cod[TAM_T];
26 }
27 } Pedido;
28
33 typedef struct
34 {
35     int qt_pedidos;
36     Pedido pedido[TAM_A];
37 }
38 } Pedidos;
39
44 typedef struct
45 {
46     char cod[TAM_T], tipo[TAM_T];
47     float custo;
48     int autonomia, id;
49 }
50 } Meio;
51
56 typedef struct
57 {
58     int qt_meios;
59     Meio meio[TAM_A];
60 }
61 } Esquema;
62
63 #endif

```

## 4.3 Funcoes.c File Reference

Neste ficheiro .c será introduzido todas as funções do programa exceto a main.

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "Funcoes.h"

```

### Functions

- int [existePedido](#) ([Pedidos](#) \*ps, int num)  
*Espaço reservado as funções de pedidos do utilizador.*
- int [insertPedido](#) ([Pedidos](#) \*ps, int id, [Pedido](#) p)  
*Inserção de um novo pedido de utilização.*
- void [criarPedido](#) ([Pedidos](#) \*ps)  
*Atribuição dos valores a serem inseridos em um novo pedido de utilização.*
- void [mostraPedidos](#) ([Pedidos](#) \*ps)  
*Listagem de todo os pedidos de utilização.*
- void [mostraPedido](#) ([Pedido](#) p)  
*Listagem de um pedido de utilização.*
- int [removerPedido](#) (char cod[], [Pedidos](#) \*ps)  
*Remoção de um pedido de utilização a partir do seu código.*

- float [custoPedido](#) (int num, [Esquema](#) \*e, [Pedidos](#) \*ps)  
*Cálculo do custo associado a um pedido de utilização a partir da indicação do seu número de ordem.*
- [Meio](#) [buscaMeio](#) ([Esquema](#) \*e, char cod[])  
*Espaço reservado as funções dos meios de mobilidade.*
- int [existeMeio](#) ([Esquema](#) \*e, char cod[])  
*verifica se um meio existe*
- int [insertMeio](#) ([Esquema](#) \*e, int id, [Meio](#) m)  
*Inserção de um novo meio de mobilidade elétrica.*
- void [criarMeio](#) ([Esquema](#) \*e)  
*Atribuição dos valores a serem inseridos em um novo meio de mobilidade elétrica.*
- void [mostraMeios](#) ([Esquema](#) \*e)  
*Listagem dos dados de todos os meios de mobilidade elétrica.*
- void [mostraMeio](#) ([Meio](#) m)  
*Listagem dos dados de um meio de mobilidade elétrica.*
- int [removerMeio](#) ([Esquema](#) \*e, char cod[])  
*Remoção de um meio de mobilidade elétrica a partir do seu código.*
- void [listarPlanoMeio](#) ([Esquema](#) \*e, [Pedidos](#) \*ps, char cod[])  
*Listagem do plano de utilização de um determinado meio de mobilidade elétrica, a partir do seu código, por ordem crescente de número de ordem do pedido.*
- char \* [procuraVeiculoSuficiente](#) ([Esquema](#) \*e, int distancia, char tipo[])  
*Espaço reservado as funções de distribuição dos meios pelos utilizadores.*
- [Pedido](#) [verificaAutonomia](#) ([Esquema](#) \*e, [Pedido](#) p)  
*verifica se a autonomia de um meio é ultrapassada pela distancia de um pedido retorn pedido com codigo alterado ou nao*
- void [distribuirMeios](#) ([Esquema](#) \*e, [Pedidos](#) \*ps)  
*realiza uma distribuição dos meios de mobilidade pelos utilizadores*
- void [interfacePrincipal](#) ()  
*Espaço reservado as funções de menus.*
- void [menuPrincipal](#) ([Esquema](#) \*e, [Pedidos](#) \*ps)  
*menu com opcoes que correspondem a interface principal*
- void [interfaceMeio](#) ()  
*parte visual do menu dos meios de mobilidade*
- void [menuMeio](#) ([Esquema](#) \*e, [Pedidos](#) \*ps)  
*menu com opcoes que correspondem a interface dos meios de mobilidade*
- void [interfacePedido](#) ()  
*parte visual do menu de pedidos do utilizador*
- void [menuPedido](#) ([Esquema](#) \*e, [Pedidos](#) \*ps)  
*menu com opcoes que correspondem a interface dos pedidos do utilizador*
- int [lerMeios](#) ([Esquema](#) \*e)  
*Espaço reservado as funções que manipulam ficheiros.*
- int [guardarMeios](#) ([Esquema](#) \*e)  
*manipula o ficheiro esquema.txt para escrita*
- int [lerPedidos](#) ([Pedidos](#) \*ps)  
*manipula o ficheiro historico.txt para leitura*
- int [guardarPedidos](#) ([Pedidos](#) \*ps)  
*manipula o ficheiro historico.txt para escrita*

### 4.3.1 Detailed Description

Neste ficheiro .c será introduzido todas as funções do programa exceto a main.

#### Author

João Apresentação ( [a21152@alunos.ipca.pt](mailto:a21152@alunos.ipca.pt) )

Gonçalo Cunha ( [a21145@alunos.ipca.pt](mailto:a21145@alunos.ipca.pt) )

Carlos Barbosa ( [a21157@alunos.ipca.pt](mailto:a21157@alunos.ipca.pt) )

#### Version

0.1

#### Date

2022-01-22

#### Copyright

Copyright (c) 2022

### 4.3.2 Function Documentation

#### 4.3.2.1 buscaMeio()

```
Meio buscaMeio (
    Esquema * e,
    char cod[] )
```

Espaço reservado as funções dos meios de mobilidade.

Procura e retorna um meio de mobilidade

#### Parameters

<i>e</i>	esquema
<i>cod</i>	codigo

#### Returns

[Meio](#)

#### 4.3.2.2 criarMeio()

```
void criarMeio (
    Esquema * e )
```

Atribuição dos valores a serem inseridos em um novo meio de mobilidade elétrica.

##### Parameters

<i>e</i>	esquema
----------	---------

#### 4.3.2.3 criarPedido()

```
void criarPedido (
    Pedidos * ps )
```

Atribuição dos valores a serem inseridos em um novo pedido de utilização.

##### Parameters

<i>ps</i>	pedidos
-----------	---------

#### 4.3.2.4 custoPedido()

```
float custoPedido (
    int num,
    Esquema * e,
    Pedidos * ps )
```

Cálculo do custo associado a um pedido de utilização a partir da indicação do seu número de ordem.

##### Parameters

<i>num</i>	numero de ordem
<i>e</i>	esquema
<i>ps</i>	pedidos

##### Returns

float

#### 4.3.2.5 distribuirMeios()

```
void distribuirMeios (
    Esquema * e,
    Pedidos * ps )
```

realiza uma distribuição dos meios de mobilidade pelos utilizadores

##### Parameters

<i>e</i>	esquema
<i>ps</i>	pedidos

#### 4.3.2.6 existeMeio()

```
int existeMeio (
    Esquema * e,
    char cod[] )
```

verifica se um meio existe

##### Parameters

<i>e</i>	esquema
<i>cod</i>	codigo

##### Returns

int

#### 4.3.2.7 existePedido()

```
int existePedido (
    Pedidos * ps,
    int num )
```

Espaço reservado as funções de pedidos do utilizador.

Verifica se um pedido existe através do seu numero de ordem

##### Parameters

<i>ps</i>	pedidos
<i>num</i>	numero de ordem



**Returns**

int

**4.3.2.8 guardarMeios()**

```
int guardarMeios (
    Esquema * e )
```

manipula o ficheiro esquema.txt para escrita

**Parameters**

<i>e</i>	esquema
----------	---------

**4.3.2.9 guardarPedidos()**

```
int guardarPedidos (
    Pedidos * ps )
```

manipula o ficheiro historico.txt para escrita

**Parameters**

<i>ps</i>	pedidos
-----------	---------

**4.3.2.10 insertMeio()**

```
int insertMeio (
    Esquema * e,
    int id,
    Meio m )
```

Inserção de um novo meio de mobilidade elétrica.

**Parameters**

<i>e</i>	esquema
<i>id</i>	identificador do esquema
<i>cod</i>	codigo
<i>tipo</i>	
<i>custo</i>	
<i>autonomia</i>	

**Returns**

int

**4.3.2.11 insertPedido()**

```
int insertPedido (
    Pedidos * ps,
    int id,
    Pedido p )
```

Inserção de um novo pedido de utilização.

**Parameters**

<i>ps</i>	pedidos
<i>id</i>	identificador do pedido
<i>nif</i>	nif do utilizador
<i>cod</i>	codigo
<i>tempo</i>	
<i>distancia</i>	

**Returns**

int

**4.3.2.12 interfacePrincipal()**

```
void interfacePrincipal ( )
```

Espaço reservado as funções de menus.

parte visual do menu principal

**4.3.2.13 lerMeios()**

```
int lerMeios (
    Esquema * e )
```

Espaço reservado as funções que manipulam ficheiros.

manipula o ficheiro esquema.txt para leitura

**Parameters**

<i>e</i>	esquema
----------	---------

**Returns**

int

**4.3.2.14 lerPedidos()**

```
int lerPedidos (
    Pedidos * ps )
```

manipula o ficheiro historico.txt para leitura

**Parameters**

<i>ps</i>	pedidos
-----------	---------

**Returns**

int

**4.3.2.15 listarPlanoMeio()**

```
void listarPlanoMeio (
    Esquema * e,
    Pedidos * ps,
    char cod[ ] )
```

Listagem do plano de utilização de um determinado meio de mobilidade elétrica, a partir do seu código, por ordem crescente de número de ordem do pedido.

**Parameters**

<i>e</i>	esquema
<i>ps</i>	pedidos
<i>cod</i>	codigo

**4.3.2.16 menuMeio()**

```
void menuMeio (
    Esquema * e,
    Pedidos * ps )
```

menu com opcoes que correspondem a interface dos meios de mobilidade

**Parameters**

<i>e</i>	
<i>ps</i>	

**4.3.2.17 menuPedido()**

```
void menuPedido (
    Esquema * e,
    Pedidos * ps )
```

menu com opcoes que correspondem a interface dos pedidos do utilizador

**Parameters**

<i>e</i>	
<i>ps</i>	

**4.3.2.18 menuPrincipal()**

```
void menuPrincipal (
    Esquema * e,
    Pedidos * ps )
```

menu com opcoes que correspondem a interface principal

**Parameters**

<i>e</i>	esquema
<i>ps</i>	pedidos

**4.3.2.19 mostraMeio()**

```
void mostraMeio (
    Meio m )
```

Listagem dos dados de um meio de mobilidade elétrica.

**Parameters**

<i>m</i>	meio de mobilidade
----------	--------------------

#### 4.3.2.20 mostraMeios()

```
void mostraMeios (
    Esquema * e )
```

Listagem dos dados de todos os meios de mobilidade elétrica.

##### Parameters

<i>e</i>	esquema
----------	---------

#### 4.3.2.21 mostraPedido()

```
void mostraPedido (
    Pedido p )
```

Listagem de um pedido de utilização.

##### Parameters

<i>p</i>	pedido
----------	--------

#### 4.3.2.22 mostraPedidos()

```
void mostraPedidos (
    Pedidos * ps )
```

Listagem de todo os pedidos de utilização.

##### Parameters

<i>ps</i>	pedidos
-----------	---------

#### 4.3.2.23 procuraVeiculoSuficiente()

```
char * procuraVeiculoSuficiente (
    Esquema * e,
    int distancia,
    char tipo[] )
```

Espaço reservado as funções de distribuição dos meios pelos utilizadores.

procura um veiculo com autonomia suficiente para uma determinada distancia a percorrer retorna o novo codigo

#### Parameters

<i>e</i>	esquema
<i>distancia</i>	
<i>tipo</i>	

#### Returns

char\*

#### 4.3.2.24 removerMeio()

```
int removerMeio (
    Esquema * e,
    char cod[] )
```

Remoção de um meio de mobilidade elétrica a partir do seu código.

#### Parameters

<i>e</i>	esquema
<i>cod</i>	codigo

#### 4.3.2.25 removerPedido()

```
int removerPedido (
    char cod[],
    Pedidos * ps )
```

Remoção de um pedido de utilização a partir do seu código.

#### Parameters

<i>cod</i>	codigo
<i>ps</i>	pedidos

#### Returns

int

## 4.3.2.26 verificaAutonomia()

```
Pedido verificaAutonomia (
    Esquema * e,
    Pedido p )
```

verifica se a autonomia de um meio é ultrapassada pela distancia de um pedido retorn pedido com codigo alterado ou nao

## Parameters

<i>e</i>	esquema
<i>p</i>	pedido

## Returns

Pedido

## 4.4 Funcoes.h File Reference

Nesta biblioteca será introduzido código referente a todas as assinaturas de funções.

```
#include <stdio.h>
#include "Estruturas.h"
```

## Functions

- int [existePedido](#) ([Pedidos](#) \*ps, int num)  
*Espaço reservado as funções de pedidos do utilizador.*
- void [mostraPedidos](#) ([Pedidos](#) \*ps)  
*Listagem de todo os pedidos de utilização.*
- void [mostraPedido](#) ([Pedido](#) p)  
*Listagem de um pedido de utilização.*
- int [guardarPedidos](#) ([Pedidos](#) \*ps)  
*manipula o ficheiro historico.txt para escrita*
- int [lerPedidos](#) ([Pedidos](#) \*ps)  
*manipula o ficheiro historico.txt para leitura*
- int [insertPedido](#) ([Pedidos](#) \*ps, int id, [Pedido](#) p)  
*Inserção de um novo pedido de utilização.*
- void [criarPedido](#) ([Pedidos](#) \*ps)  
*Atribuição dos valores a serem inseridos em um novo pedido de utilização.*
- int [removerPedido](#) (char cod[], [Pedidos](#) \*ps)  
*Remoção de um pedido de utilização a partir do seu código.*
- float [custoPedido](#) (int num, [Esquema](#) \*e, [Pedidos](#) \*ps)  
*Cálculo do custo associado a um pedido de utilização a partir da indicação do seu número de ordem.*
- int [existeMeio](#) ([Esquema](#) \*e, char cod[])  
*verifica se um meio existe*
- void [mostraMeios](#) ([Esquema](#) \*e)

- Listagem dos dados de todos os meios de mobilidade elétrica.*

  - void **mostraMeio** (**Meio** m)
- Listagem dos dados de um meio de mobilidade elétrica.*

  - int **insertMeio** (**Esquema** \*e, int count, **Meio** m)
- Inserção de um novo meio de mobilidade elétrica.*

  - int **lerMeios** (**Esquema** \*e)
- Espaço reservado as funções que manipulam ficheiros.*

  - int **guardarMeios** (**Esquema** \*e)

*manipula o ficheiro esquema.txt para escrita*
- void **criarMeio** (**Esquema** \*e)

*Atribuição dos valores a serem inseridos em um novo meio de mobilidade elétrica.*
- int **removerMeio** (**Esquema** \*e, char cod[])

*Remoção de um meio de mobilidade elétrica a partir do seu código.*
- void **listarPlanoMeio** (**Esquema** \*e, **Pedidos** \*ps, char cod[])

*Listagem do plano de utilização de um determinado meio de mobilidade elétrica, a partir do seu código, por ordem crescente de número de ordem do pedido.*
- **Pedido verificaAutonomia** (**Esquema** \*e, **Pedido** p)

*verifica se a autonomia de um meio é ultrapassada pela distancia de um pedido retorn pedido com codigo alterado ou nao*
- char \* **procuraVeiculoSuficiente** (**Esquema** \*e, int distancia, char tipo[])

*Espaço reservado as funções de distribuição dos meios pelos utilizadores.*
- void **distribuirMeios** (**Esquema** \*e, **Pedidos** \*ps)

*realiza uma distribuição dos meios de mobilidade pelos utilizadores*
- void **interfacePrincipal** ()

*Espaço reservado as funções de menus.*
- void **menuPrincipal** (**Esquema** \*e, **Pedidos** \*ps)

*menu com opcoes que correspondem a interface principal*
- void **interfaceMeio** ()

*parte visual do menu dos meios de mobilidade*
- void **menuMeio** (**Esquema** \*e, **Pedidos** \*ps)

*menu com opcoes que correspondem a interface dos meios de mobilidade*
- void **interfacePedido** ()

*parte visual do menu de pedidos do utilizador*
- void **menuPedido** (**Esquema** \*e, **Pedidos** \*ps)

*menu com opcoes que correspondem a interface dos pedidos do utilizador*

#### 4.4.1 Detailed Description

Nesta biblioteca será introduzido código referente a todas as assinaturas de funções.

##### Author

João Apresentação ( [a21152@alunos.ipca.pt](mailto:a21152@alunos.ipca.pt))  
 Gonçalo Cunha ( [a21145@alunos.ipca.pt](mailto:a21145@alunos.ipca.pt))  
 Carlos Barbosa ( [a21157@alunos.ipca.pt](mailto:a21157@alunos.ipca.pt))

##### Version

0.1



## Date

2022-01-22

## Copyright

Copyright (c) 2022

## 4.4.2 Function Documentation

### 4.4.2.1 criarMeio()

```
void criarMeio (
    Esquema * e )
```

Atribuição dos valores a serem inseridos em um novo meio de mobilidade elétrica.

## Parameters

<i>e</i>	esquema
----------	---------

### 4.4.2.2 criarPedido()

```
void criarPedido (
    Pedidos * ps )
```

Atribuição dos valores a serem inseridos em um novo pedido de utilização.

## Parameters

<i>ps</i>	pedidos
-----------	---------

### 4.4.2.3 custoPedido()

```
float custoPedido (
    int num,
    Esquema * e,
    Pedidos * ps )
```

Cálculo do custo associado a um pedido de utilização a partir da indicação do seu número de ordem.

**Parameters**

<i>num</i>	numero de ordem
<i>e</i>	esquema
<i>ps</i>	pedidos

**Returns**

float

**4.4.2.4 distribuirMeios()**

```
void distribuirMeios (
    Esquema * e,
    Pedidos * ps )
```

realiza uma distribuição dos meios de mobilidade pelos utilizadores

**Parameters**

<i>e</i>	esquema
<i>ps</i>	pedidos

**4.4.2.5 existeMeio()**

```
int existeMeio (
    Esquema * e,
    char cod[] )
```

verifica se um meio existe

**Parameters**

<i>e</i>	esquema
<i>cod</i>	codigo

**Returns**

int

#### 4.4.2.6 existePedido()

```
int existePedido (
    Pedidos * ps,
    int num )
```

Espaço reservado as funções de pedidos do utilizador.

Verifica se um pedido existe através do seu numero de ordem

##### Parameters

<i>ps</i>	pedidos
<i>num</i>	numero de ordem

##### Returns

int

#### 4.4.2.7 guardarMeios()

```
int guardarMeios (
    Esquema * e )
```

manipula o ficheiro esquema.txt para escrita

##### Parameters

<i>e</i>	esquema
----------	---------

#### 4.4.2.8 guardarPedidos()

```
int guardarPedidos (
    Pedidos * ps )
```

manipula o ficheiro historico.txt para escrita

##### Parameters

<i>ps</i>	pedidos
-----------	---------

#### 4.4.2.9 insertMeio()

```
int insertMeio (
    Esquema * e,
    int id,
    Meio m )
```

Inserção de um novo meio de mobilidade elétrica.

##### Parameters

<i>e</i>	esquema
<i>id</i>	identificador do esquema
<i>cod</i>	codigo
<i>tipo</i>	
<i>custo</i>	
<i>autonomia</i>	

##### Returns

int

#### 4.4.2.10 insertPedido()

```
int insertPedido (
    Pedidos * ps,
    int id,
    Pedido p )
```

Inserção de um novo pedido de utilização.

##### Parameters

<i>ps</i>	pedidos
<i>id</i>	identificador do pedido
<i>nif</i>	nif do utilizador
<i>cod</i>	codigo
<i>tempo</i>	
<i>distancia</i>	

##### Returns

int

#### 4.4.2.11 interfacePrincipal()

```
void interfacePrincipal ( )
```

Espaço reservado as funções de menus.

parte visual do menu principal

#### 4.4.2.12 lerMeios()

```
int lerMeios (
    Esquema * e )
```

Espaço reservado as funções que manipulam ficheiros.

manipula o ficheiro esquema.txt para leitura

##### Parameters

<i>e</i>	esquema
----------	---------

##### Returns

int

#### 4.4.2.13 lerPedidos()

```
int lerPedidos (
    Pedidos * ps )
```

manipula o ficheiro historico.txt para leitura

##### Parameters

<i>ps</i>	pedidos
-----------	---------

##### Returns

int

#### 4.4.2.14 listarPlanoMeio()

```
void listarPlanoMeio (
    Esquema * e,
```

```
Pedidos * ps,  
char cod[ ] )
```

Listagem do plano de utilização de um determinado meio de mobilidade elétrica, a partir do seu código, por ordem crescente de número de ordem do pedido.

#### Parameters

<i>e</i>	esquema
<i>ps</i>	pedidos
<i>cod</i>	codigo

#### 4.4.2.15 menuMeio()

```
void menuMeio (   
    Esquema * e,   
    Pedidos * ps )
```

menu com opcoes que correspondem a interface dos meios de mobilidade

#### Parameters

<i>e</i>	
<i>ps</i>	

#### 4.4.2.16 menuPedido()

```
void menuPedido (   
    Esquema * e,   
    Pedidos * ps )
```

menu com opcoes que correspondem a interface dos pedidos do utilizador

#### Parameters

<i>e</i>	
<i>ps</i>	

#### 4.4.2.17 menuPrincipal()

```
void menuPrincipal (   
    Esquema * e,   
    Pedidos * ps )
```

menu com opcoes que correspondem a interface principal

## Parameters

<i>e</i>	esquema
<i>ps</i>	pedidos

**4.4.2.18 mostraMeio()**

```
void mostraMeio (  
    Meio m )
```

Listagem dos dados de um meio de mobilidade elétrica.

## Parameters

<i>m</i>	meio de mobilidade
----------	--------------------

**4.4.2.19 mostraMeios()**

```
void mostraMeios (  
    Esquema * e )
```

Listagem dos dados de todos os meios de mobilidade elétrica.

## Parameters

<i>e</i>	esquema
----------	---------

**4.4.2.20 mostraPedido()**

```
void mostraPedido (  
    Pedido p )
```

Listagem de um pedido de utilização.

## Parameters

<i>p</i>	pedido
----------	--------



#### 4.4.2.21 mostraPedidos()

```
void mostraPedidos (
    Pedidos * ps )
```

Listagem de todo os pedidos de utilização.

##### Parameters

<i>ps</i>	pedidos
-----------	---------

#### 4.4.2.22 procuraVeiculoSuficiente()

```
char * procuraVeiculoSuficiente (
    Esquema * e,
    int distancia,
    char tipo[] )
```

Espaço reservado as funções de distribuição dos meios pelos utilizadores.

procura um veiculo com autonomia suficiente para uma determinada distancia a percorrer retorna o novo codigo

##### Parameters

<i>e</i>	esquema
<i>distancia</i>	
<i>tipo</i>	

##### Returns

char\*

#### 4.4.2.23 removerMeio()

```
int removerMeio (
    Esquema * e,
    char cod[] )
```

Remoção de um meio de mobilidade elétrica a partir do seu código.

##### Parameters

<i>e</i>	esquema
<i>cod</i>	codigo

#### 4.4.2.24 removerPedido()

```
int removerPedido (
    char cod[],
    Pedidos * ps )
```

Remoção de um pedido de utilização a partir do seu código.

##### Parameters

<i>cod</i>	codigo
<i>ps</i>	pedidos

##### Returns

int

#### 4.4.2.25 verificaAutonomia()

```
Pedido verificaAutonomia (
    Esquema * e,
    Pedido p )
```

verifica se a autonomia de um meio é ultrapassada pela distancia de um pedido retorn pedido com codigo alterado ou nao

##### Parameters

<i>e</i>	esquema
<i>p</i>	pedido

##### Returns

Pedido

## 4.5 Funcoes.h

[Go to the documentation of this file.](#)

```
1
14 #ifndef FUNCOES
15 #define FUNCOES
16 #include <stdio.h>
17 #include "Estruturas.h"
18
19 #pragma region ASSINATURAS PEDIDO
20 int existePedido(Pedidos *ps, int num);
21 void mostraPedidos(Pedidos *ps);
```

```

22 void mostraPedido(Pedido p);
23 int guardarPedidos(Pedidos *ps);
24 int lerPedidos(Pedidos *ps);
25 int insertPedido(Pedidos *ps, int id, Pedido p);
26 void criarPedido(Pedidos *ps);
27 int removerPedido(char cod[], Pedidos *ps);
28 float custoPedido(int num, Esquema *e, Pedidos *ps);
29 #pragma endregion
30
31 #pragma region ASSINATURAS MEIO
32 int existeMeio(Esquema *e, char cod[]);
33 void mostraMeios(Esquema *e);
34 void mostraMeio(Meio m);
35 int insertMeio(Esquema *e, int count, Meio m);
36 int lerMeios(Esquema *e);
37 int guardarMeios(Esquema *e);
38 void criarMeio(Esquema *e);
39 int removerMeio(Esquema *e, char cod[]);
40 void listarPlanoMeio(Esquema *e, Pedidos *ps, char cod[]);
41 #pragma endregion
42
43 #pragma region ASSISNATURAS AUTONOMIA
44 Pedido verificaAutonomia(Esquema *e, Pedido p);
45 char *procuraVeiculoSuficiente(Esquema *e, int distancia, char tipo[]);
46 void distribuirMeios(Esquema *e, Pedidos *ps);
47 #pragma endregion
48
49 #pragma region ASSINATURAS MENU
50 void interfacePrincipal();
51 void menuPrincipal(Esquema *e, Pedidos *ps);
52 void interfaceMeio();
53 void menuMeio(Esquema *e, Pedidos *ps);
54 void interfacePedido();
55 void menuPedido(Esquema *e, Pedidos *ps);
56 #pragma endregion
57
58 #endif

```

## 4.6 main.c File Reference

Neste ficheiro .c apenas terá a função main para execução do programa.

```

#include <stdio.h>
#include "Funcoes.h"

```

### Functions

- int [main](#) ()  
*função main*

#### 4.6.1 Detailed Description

Neste ficheiro .c apenas terá a função main para execução do programa.

#### Author

João Apresentação ( [a21152@alunos.ipca.pt](mailto:a21152@alunos.ipca.pt))  
 Gonçalo Cunha ( [a21145@alunos.ipca.pt](mailto:a21145@alunos.ipca.pt))  
 Carlos Barbosa ( [a21157@alunos.ipca.pt](mailto:a21157@alunos.ipca.pt))

**Version**

0.1

**Date**

2022-01-20

**Copyright**

Copyright (c) 2022

## 4.6.2 Function Documentation

### 4.6.2.1 main()

```
int main ( )
```

função main

**Returns**

int

# Index

buscaMeio  
    Funcoes.c, 10

criarMeio  
    Funcoes.c, 10  
    Funcoes.h, 21

criarPedido  
    Funcoes.c, 11  
    Funcoes.h, 21

custoPedido  
    Funcoes.c, 11  
    Funcoes.h, 21

distribuirMeios  
    Funcoes.c, 11  
    Funcoes.h, 22

Esquema, 5

Estruturas.h, 7

existeMeio  
    Funcoes.c, 12  
    Funcoes.h, 22

existePedido  
    Funcoes.c, 12  
    Funcoes.h, 22

Funcoes.c, 8  
    buscaMeio, 10  
    criarMeio, 10  
    criarPedido, 11  
    custoPedido, 11  
    distribuirMeios, 11  
    existeMeio, 12  
    existePedido, 12  
    guardarMeios, 13  
    guardarPedidos, 13  
    insertMeio, 13  
    insertPedido, 14  
    interfacePrincipal, 14  
    lerMeios, 14  
    lerPedidos, 15  
    listarPlanoMeio, 15  
    menuMeio, 15  
    menuPedido, 16  
    menuPrincipal, 16  
    mostraMeio, 16  
    mostraMeios, 17  
    mostraPedido, 17  
    mostraPedidos, 17  
    procuraVeiculoSuficiente, 17

removerMeio, 18  
removerPedido, 18  
verificaAutonomia, 18

Funcoes.h, 19  
    criarMeio, 21  
    criarPedido, 21  
    custoPedido, 21  
    distribuirMeios, 22  
    existeMeio, 22  
    existePedido, 22  
    guardarMeios, 23  
    guardarPedidos, 23  
    insertMeio, 23  
    insertPedido, 24  
    interfacePrincipal, 24  
    lerMeios, 25  
    lerPedidos, 25  
    listarPlanoMeio, 25  
    menuMeio, 26  
    menuPedido, 26  
    menuPrincipal, 26  
    mostraMeio, 28  
    mostraMeios, 28  
    mostraPedido, 28  
    mostraPedidos, 28  
    procuraVeiculoSuficiente, 29  
    removerMeio, 29  
    removerPedido, 30  
    verificaAutonomia, 30

guardarMeios  
    Funcoes.c, 13  
    Funcoes.h, 23

guardarPedidos  
    Funcoes.c, 13  
    Funcoes.h, 23

insertMeio  
    Funcoes.c, 13  
    Funcoes.h, 23

insertPedido  
    Funcoes.c, 14  
    Funcoes.h, 24

interfacePrincipal  
    Funcoes.c, 14  
    Funcoes.h, 24

lerMeios  
    Funcoes.c, 14  
    Funcoes.h, 25

- lerPedidos
  - Funcoes.c, [15](#)
  - Funcoes.h, [25](#)
- listarPlanoMeio
  - Funcoes.c, [15](#)
  - Funcoes.h, [25](#)
- main
  - main.c, [32](#)
- main.c, [31](#)
  - main, [32](#)
- Meio, [5](#)
- menuMeio
  - Funcoes.c, [15](#)
  - Funcoes.h, [26](#)
- menuPedido
  - Funcoes.c, [16](#)
  - Funcoes.h, [26](#)
- menuPrincipal
  - Funcoes.c, [16](#)
  - Funcoes.h, [26](#)
- mostraMeio
  - Funcoes.c, [16](#)
  - Funcoes.h, [28](#)
- mostraMeios
  - Funcoes.c, [17](#)
  - Funcoes.h, [28](#)
- mostraPedido
  - Funcoes.c, [17](#)
  - Funcoes.h, [28](#)
- mostraPedidos
  - Funcoes.c, [17](#)
  - Funcoes.h, [28](#)
- Pedido, [6](#)
- Pedidos, [6](#)
- procuraVeiculoSuficiente
  - Funcoes.c, [17](#)
  - Funcoes.h, [29](#)
- removerMeio
  - Funcoes.c, [18](#)
  - Funcoes.h, [29](#)
- removerPedido
  - Funcoes.c, [18](#)
  - Funcoes.h, [30](#)
- verificaAutonomia
  - Funcoes.c, [18](#)
  - Funcoes.h, [30](#)