

Projeto Gestão de Antenas

Estruturas de Dados Avançadas

Licenciatura em Engenharia de Sistemas Informáticos

Autor: Ana Rita Ranito

Professor: Luís Ferreira

1 Índice dos componentes	1
1.1 Lista de componentes	1
2 Índice dos ficheiros	3
2.1 Lista de ficheiros	3
3 Documentação da classe	5
3.1 Referência à estrutura Antena	5
3.1.1 Descrição detalhada	5
3.1.2 Documentação dos dados membro	5
3.1.2.1 frequencia	5
3.1.2.2 proxima	6
3.1.2.3 y	6
3.2 Referência à estrutura AntenaBinaria	6
3.2.1 Descrição detalhada	6
3.2.2 Documentação dos dados membro	6
3.2.2.1 frequencia	6
3.2.2.2 y	6
4 Documentação do ficheiro	7
4.1 Referência ao ficheiro estruturas.h	7
4.1.1 Descrição detalhada	8
4.2 estruturas.h	8
4.3 Referência ao ficheiro funcoes.c	8
4.3.1 Descrição detalhada	9
4.3.2 Documentação das funções	9
4.3.2.1 calcularEdesenharMatriz()	9
4.3.2.2 carregarDeFicheiro()	10
4.3.2.3 criarAntena()	10
4.3.2.4 exhibirAntenas()	10
4.3.2.5 gravarEmBinario()	10
4.3.2.6 inserirAntena()	10
4.3.2.7 removerAntena()	11
4.4 Referência ao ficheiro funcoes.h	11
4.4.1 Descrição detalhada	12
4.4.2 Documentação das funções	12
4.4.2.1 calcularEdesenharMatriz()	12
4.4.2.2 carregarDeFicheiro()	13
4.4.2.3 criarAntena()	13
4.4.2.4 exhibirAntenas()	13
4.4.2.5 gravarEmBinario()	14
4.4.2.6 inserirAntena()	14
4.4.2.7 removerAntena()	14

4.5 funcoes.h	15
4.6 Referência ao ficheiro main.c	15
4.6.1 Descrição detalhada	16
4.6.2 Documentação das funções	16
4.6.2.1 main()	16
Índice	17

Capítulo 1

Índice dos componentes

1.1 Lista de componentes

Lista de classes, estruturas, uniões e interfaces com uma breve descrição:

Antena	Estrutura que representa uma antena com frequência e coordenadas	5
AntenaBinaria	Estrutura auxiliar usada para gravar ou ler dados binários de uma antena. Esta estrutura contém apenas os campos essenciais: frequência e coordenadas (x e y)	6

Capítulo 2

Índice dos ficheiros

2.1 Lista de ficheiros

Lista de todos os ficheiros documentados com uma breve descrição:

estruturas.h	7
funcoes.c	8
funcoes.h	11
main.c	15

Capítulo 3

Documentação da classe

3.1 Referência à estrutura Antena

Estrutura que representa uma antena com frequência e coordenadas.

```
#include <estruturas.h>
```

Diagrama de colaboração para Antena:



Atributos Públicos

- char `frequencia`
- int `x`
- int `y`
- struct `Antena` * `proxima`

3.1.1 Descrição detalhada

Estrutura que representa uma antena com frequência e coordenadas.

3.1.2 Documentação dos dados membro

3.1.2.1 `frequencia`

```
char Antena::frequencia
```

Frequência da antena.

3.1.2.2 proxima

```
struct Antena* Antena::proxima
```

Ponteiro para a próxima antena na lista.

3.1.2.3 y

```
int Antena::y
```

Coordenadas (coluna x, linha y).

A documentação para esta estrutura foi gerada a partir do seguinte ficheiro:

- [estruturas.h](#)

3.2 Referência à estrutura AntenaBinaria

Estrutura auxiliar usada para gravar ou ler dados binários de uma antena. Esta estrutura contém apenas os campos essenciais: frequência e coordenadas (x e y).

```
#include <estruturas.h>
```

Atributos Públicos

- char [frequencia](#)
- int [x](#)
- int [y](#)

3.2.1 Descrição detalhada

Estrutura auxiliar usada para gravar ou ler dados binários de uma antena. Esta estrutura contém apenas os campos essenciais: frequência e coordenadas (x e y).

3.2.2 Documentação dos dados membro

3.2.2.1 frequencia

```
char AntenaBinaria::frequencia
```

Frequência da antena

3.2.2.2 y

```
int AntenaBinaria::y
```

Coordenadas da antena

A documentação para esta estrutura foi gerada a partir do seguinte ficheiro:

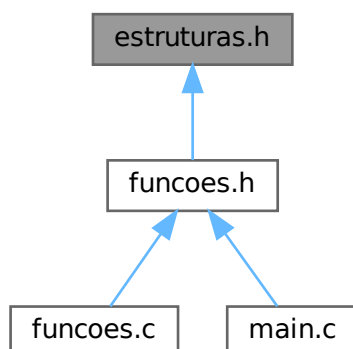
- [estruturas.h](#)

Capítulo 4

Documentação do ficheiro

4.1 Referência ao ficheiro estruturas.h

Este grafo mostra quais são os ficheiros que incluem directamente ou indirectamente este ficheiro:



Componentes

- struct [Antena](#)

Estrutura que representa uma antena com frequência e coordenadas.

- struct [AntenaBinaria](#)

Estrutura auxiliar usada para gravar ou ler dados binários de uma antena. Esta estrutura contém apenas os campos essenciais: frequência e coordenadas (x e y).

Definições de tipos

- typedef struct [Antena](#) **Antena**

Estrutura que representa uma antena com frequência e coordenadas.

4.1.1 Descrição detalhada

Autor

Rita Ranito

Data

2025-03-30 @project Projeto Gestão de Antenas

Declaração de estruturas.

4.2 estruturas.h

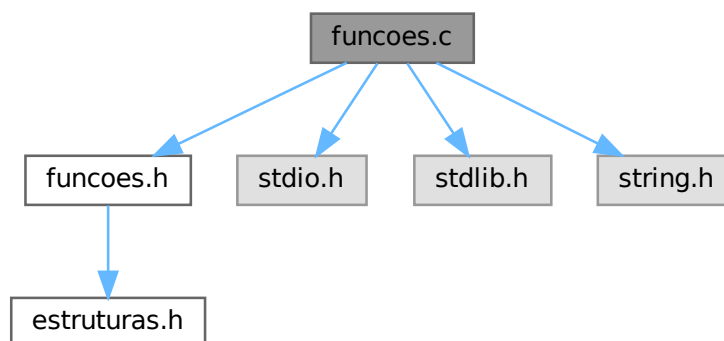
[Ir para a documentação deste ficheiro.](#)

```
00001
00010 #ifndef ESTRUTURAS_H
00011 #define ESTRUTURAS_H
00012
00016 typedef struct Antena {
00017     char frequencia;
00018     int x, y;
00019     struct Antena *proxima;
00020 } Antena;
00021
00026 typedef struct {
00027     char frequencia;
00028     int x, y;
00029 } AntenaBinaria;
00030
00031
00032
00033
00034 #endif
```

4.3 Referência ao ficheiro funcoes.c

```
#include "funcoes.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Diagrama de dependências de inclusão para funcoes.c:



Funções

- **Antena** * **criarAntena** (char frequencia, int x, int y)
Cria uma nova antena com os dados fornecidos.
- void **inserirAntena** (**Antena** **cabeca, char frequencia, int x, int y)
Insere uma nova antena no início da lista ligada.
- void **removerAntena** (**Antena** **cabeca, char frequencia, int x, int y)
Remove uma antena com coordenadas e frequência específicas.
- **Antena** * **carregarDeFicheiro** (const char *nomeFicheiro, int *linhas, int *colunas)
Carrega uma lista de antenas a partir de um ficheiro de texto.
- void **exibirAntenas** (**Antena** *cabeca)
Exibe a lista de antenas no terminal.
- void **calcularEdesenharMatriz** (**Antena** *cabeca, int linhas, int colunas)
Cria e imprime uma matriz com antenas e interferências calculadas.
- void **gravarEmBinario** (**Antena** *cabeca, const char *nomeFicheiro)
Grava a lista de antenas num ficheiro binário.

4.3.1 Descrição detalhada

Autor

Rita Ranito

Data

2025-03-30 @project Projeto Gestão de Antenas

Implementação funções.

4.3.2 Documentação das funções

4.3.2.1 calcularEdesenharMatriz()

```
void calcularEdesenharMatriz (  
    Antena * cabeca,  
    int linhas,  
    int colunas )
```

Cria e imprime uma matriz com antenas e interferências calculadas.

Calcula e desenha a matriz com antenas e suas interferências.

Esta função inicializa uma matriz com pontos vazios ('.'), posiciona as antenas com base nas coordenadas armazenadas na lista ligada, e calcula os pontos de interferência entre antenas da mesma frequência, marcando-os com '#'. No final, imprime a matriz no terminal.

4.3.2.2 carregarDeFicheiro()

```
Antena * carregarDeFicheiro (
    const char * nomeFicheiro,
    int * linhas,
    int * colunas )
```

Carrega uma lista de antenas a partir de um ficheiro de texto.

Lê um ficheiro de texto linha a linha, criando antenas para cada caracter diferente de '.'.

4.3.2.3 criarAntena()

```
Antena * criarAntena (
    char frequencia,
    int x,
    int y )
```

Cria uma nova antena com os dados fornecidos.

Cria dinamicamente uma nova antena com os dados fornecidos.

4.3.2.4 exibirAntenas()

```
void exibirAntenas (
    Antena * cabeca )
```

Exibe a lista de antenas no terminal.

Mostra no terminal todas as antenas guardadas na lista.

4.3.2.5 gravarEmBinario()

```
void gravarEmBinario (
    Antena * cabeca,
    const char * nomeFicheiro )
```

Grava a lista de antenas num ficheiro binário.

Parâmetros

<i>cabeca</i>	Ponteiro para a cabeça da lista ligada de antenas.
<i>nomeFicheiro</i>	Nome do ficheiro binário onde os dados serão guardados.

4.3.2.6 inserirAntena()

```
void inserirAntena (
    Antena ** cabeca,
```

```
char frequencia,  
int x,  
int y )
```

Insere uma nova antena no início da lista ligada.

Insere uma nova antena no início da lista ligada.

4.3.2.7 removerAntena()

```
void removerAntena (  
    Antena ** cabeca,  
    char frequencia,  
    int x,  
    int y )
```

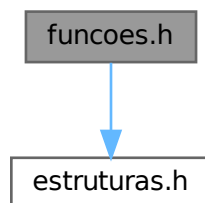
Remove uma antena com coordenadas e frequência específicas.

Procura e remove uma antena com frequência e coordenadas específicas.

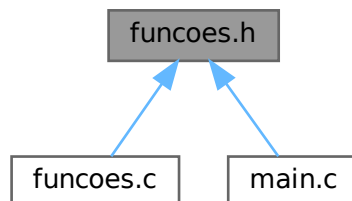
4.4 Referência ao ficheiro funcoes.h

```
#include "estruturas.h"
```

Diagrama de dependências de inclusão para funcoes.h:



Este grafo mostra quais são os ficheiros que incluem directamente ou indirectamente este ficheiro:



Funções

- **Antena * criarAntena** (char frequencia, int x, int y)
Cria uma nova antena com os dados fornecidos.
- void **inserirAntena** (**Antena **cabeca**, char frequencia, int x, int y)
Insere uma nova antena no início da lista ligada.
- void **removerAntena** (**Antena **cabeca**, char frequencia, int x, int y)
Remove uma antena com coordenadas e frequência específicas.
- **Antena * carregarDeFicheiro** (const char *nomeFicheiro, int *linhas, int *colunas)
Carrega uma lista de antenas a partir de um ficheiro de texto.
- void **exibirAntenas** (**Antena *cabeca**)
Exibe a lista de antenas no terminal.
- void **calcularEdesenharMatriz** (**Antena *cabeca**, int linhas, int colunas)
Calcula e desenha a matriz com antenas e suas interferências.
- void **gravarEmBinario** (**Antena *cabeca**, const char *nomeFicheiro)
Grava a lista de antenas num ficheiro binário.

4.4.1 Descrição detalhada

Autor

Rita Ranito

Data

2025-03-30 @project Projeto Gestão de Antenas

Protótipos das funções.

4.4.2 Documentação das funções

4.4.2.1 calcularEdesenharMatriz()

```
void calcularEdesenharMatriz (
    Antena * cabeca,
    int linhas,
    int colunas )
```

Calcula e desenha a matriz com antenas e suas interferências.

Parâmetros

<i>cabeca</i>	Ponteiro para a cabeça da lista de antenas.
<i>linhas</i>	Número total de linhas da matriz.
<i>colunas</i>	Número total de colunas da matriz.

Calcula e desenha a matriz com antenas e suas interferências.

Esta função inicializa uma matriz com pontos vazios ('.'), posiciona as antenas com base nas coordenadas armazenadas na lista ligada, e calcula os pontos de interferência entre antenas da mesma frequência, marcando-os com

'#'. No final, imprime a matriz no terminal.

4.4.2.2 carregarDeFicheiro()

```
Antena * carregarDeFicheiro (
    const char * nomeFicheiro,
    int * linhas,
    int * colunas )
```

Carrega uma lista de antenas a partir de um ficheiro de texto.

Parâmetros

<i>nomeFicheiro</i>	Nome do ficheiro a ler.
<i>linhas</i>	Ponteiro para armazenar o número de linhas lidas.
<i>colunas</i>	Ponteiro para armazenar o número de colunas lidas.

Retorna

Ponteiro para a lista ligada criada.

Lê um ficheiro de texto linha a linha, criando antenas para cada caracter diferente de '.'.

4.4.2.3 criarAntena()

```
Antena * criarAntena (
    char frequencia,
    int x,
    int y )
```

Cria uma nova antena com os dados fornecidos.

Parâmetros

<i>frequencia</i>	Letra que representa a frequência da antena.
<i>x</i>	Coordenada horizontal (coluna).
<i>y</i>	Coordenada vertical (linha).

Retorna

Ponteiro para a nova antena criada.

Cria dinamicamente uma nova antena com os dados fornecidos.

4.4.2.4 exhibirAntenas()

```
void exhibirAntenas (
    Antena * cabeca )
```

Exibe a lista de antenas no terminal.

Parâmetros

<i>cabeca</i>	Ponteiro para a cabeça da lista de antenas.
---------------	---

Mostra no terminal todas as antenas guardadas na lista.

4.4.2.5 gravarEmBinario()

```
void gravarEmBinario (
    Antena * cabeca,
    const char * nomeFicheiro )
```

Grava a lista de antenas num ficheiro binário.

Parâmetros

<i>cabeca</i>	Ponteiro para a cabeça da lista ligada de antenas.
<i>nomeFicheiro</i>	Nome do ficheiro binário onde os dados serão guardados.

4.4.2.6 inserirAntena()

```
void inserirAntena (
    Antena ** cabeca,
    char frequencia,
    int x,
    int y )
```

Insere uma nova antena no início da lista ligada.

Parâmetros

<i>cabeca</i>	Ponteiro para o ponteiro da cabeça da lista.
<i>frequencia</i>	Letra que representa a frequência da antena.
<i>x</i>	Coordenada horizontal (coluna).
<i>y</i>	Coordenada vertical (linha).

Insere uma nova antena no início da lista ligada.

4.4.2.7 removerAntena()

```
void removerAntena (
    Antena ** cabeca,
    char frequencia,
    int x,
    int y )
```

Remove uma antena com coordenadas e frequência específicas.

Parâmetros

<i>cabeca</i>	Ponteiro para o ponteiro da cabeça da lista.
<i>frequencia</i>	Letra da frequência a remover.
<i>x</i>	Coordenada horizontal (coluna).
<i>y</i>	Coordenada vertical (linha).

Procura e remove uma antena com frequência e coordenadas específicas.

4.5 funcoes.h

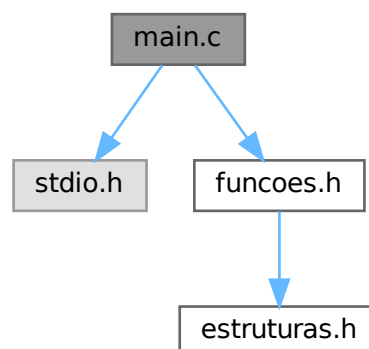
[Ir para a documentação deste ficheiro.](#)

```
00001
00010 #ifndef FUNCOES_H
00011 #define FUNCOES_H
00012 #include "estruturas.h"
00013
00021 Antena* criarAntena(char frequencia, int x, int y);
00022
00030 void inserirAntena(Antena **cabeca, char frequencia, int x, int y);
00031
00039 void removerAntena(Antena **cabeca, char frequencia, int x, int y);
00040
00048 Antena* carregarDeFicheiro(const char* nomeFicheiro, int *linhas, int *colunas);
00049
00054 void exibirAntenas(Antena *cabeca);
00055
00062 void calcularEdesenharMatriz(Antena *cabeca, int linhas, int colunas);
00063
00064
00071 void gravarEmBinario(Antena *cabeca, const char *nomeFicheiro);
00072
00073
00074 #endif
```

4.6 Referência ao ficheiro main.c

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

Diagrama de dependências de inclusão para main.c:



Funções

- `int main ()`

4.6.1 Descrição detalhada

Autor

Rita Ranito

Data

2025-03-30 @project Projeto Gestão de Antenas

Entrada principal do programa.

4.6.2 Documentação das funções

4.6.2.1 main()

```
int main ( )
```

Função principal: lê o ficheiro, carrega antenas, exhibe e desenha a matriz.

Índice

Antena, 5
 frequencia, 5
 proxima, 5
 y, 6
AntenaBinaria, 6
 frequencia, 6
 y, 6

calcularEdesenharMatriz
 funcoes.c, 9
 funcoes.h, 12
carregarDeFicheiro
 funcoes.c, 9
 funcoes.h, 13
criarAntena
 funcoes.c, 10
 funcoes.h, 13

estruturas.h, 7
exibirAntenas
 funcoes.c, 10
 funcoes.h, 13

frequencia
 Antena, 5
 AntenaBinaria, 6
funcoes.c, 8
 calcularEdesenharMatriz, 9
 carregarDeFicheiro, 9
 criarAntena, 10
 exibirAntenas, 10
 gravarEmBinario, 10
 inserirAntena, 10
 removerAntena, 11
funcoes.h, 11
 calcularEdesenharMatriz, 12
 carregarDeFicheiro, 13
 criarAntena, 13
 exibirAntenas, 13
 gravarEmBinario, 14
 inserirAntena, 14
 removerAntena, 14

gravarEmBinario
 funcoes.c, 10
 funcoes.h, 14

inserirAntena
 funcoes.c, 10
 funcoes.h, 14

main
 main.c, 16
main.c, 15
 main, 16

proxima
 Antena, 5

removerAntena
 funcoes.c, 11
 funcoes.h, 14

y
 Antena, 6
 AntenaBinaria, 6