



Instituto Politécnico do Cávado e do Ave

Escola Superior de Tecnologia

LICENCIATURA EM ENGENHARIA DE SISTEMAS
INFORMÁTICOS - PÓS LABORAL

Estrutura de Dados Avançadas

Matheus da Rocha Delgado, nº31542

Docente: Luis Gonzaga Martins Ferreira
March 30, 2025

Índice

1	Introdução	3
1.1	Problema	3
1.2	Enquadramento	3
1.3	Objetivos	3
2	Análise e Especificação	4
2.1	Estruturas de Dados Utilizadas	4
2.1.1	Estrutura Antena	4
2.1.2	Estrutura Nefasto	4
2.2	Dados a Entrar e Preservação em Ficheiros	5
2.3	Funcionalidades e Requisitos	5
2.3.1	Funcionalidades Implementadas	5
2.3.2	Especificação dos Dados de Entrada e Saída	5
3	Implementação em Linguagem C	6
3.1	Estruturas e Definições	6
3.1.1	Estrutura Antena	6
3.1.2	Estrutura Nefasto	7
3.2	Organização do Código	7
3.3	Implementação das Funções	7
3.3.1	Funções de Gestão de Antenas	7
3.4	Funções para os Efeitos Nefastos	8
3.5	Gestão de Ficheiros	8
3.6	Cálculos e Lógica dos Efeitos Nefastos	9
4	Conclusão	9
5	Referências	10

1 Introdução

1.1 Problema

No contexto da modelação de uma cidade, o sistema proposto visa gerir antenas operantes em diversas frequências e determinar os efeitos nefastos resultantes da posição relativa de pares de antenas. O efeito nefasto ocorre quando duas antenas, que operam na mesma frequência, estão posicionadas de forma a que uma delas se encontre duas vezes mais afastada que a outra, o que resulta num alinhamento que provoca um impacto negativo numa determinada localização. Este fenómeno constitui o núcleo do problema, exigindo uma abordagem algorítmica capaz de identificar e armazenar tais localizações.

1.2 Enquadramento

O desenvolvimento deste projeto insere-se no âmbito da disciplina de Estruturas de Dados Avançadas, enfatizando a importância dos apontadores e da gestão dinâmica de memória na resolução de problemas de dimensão média. O sistema utiliza listas ligadas para representar as antenas e os efeitos nefastos, demonstrando a aplicabilidade de estruturas de dados dinâmicas na resolução de desafios complexos. A modularização do código e a documentação com Doxygen constituem aspetos fundamentais, permitindo uma melhor manutenção e compreensão do sistema por parte de qualquer leitor, independentemente do seu nível de conhecimento em C ou na área de estruturas de dados.

1.3 Objetivos

Consolidar o conhecimento teórico: Aplicar os conceitos de apontadores e gestão de memória vistos em aula.

Implementar estruturas de dados dinâmicas: Utilizar listas ligadas para a gestão de antenas e dos efeitos nefastos.

Garantir modularidade e clareza do código: Separar a implementação em ficheiros distintos e documentá-la utilizando Doxygen.

Facilitar a manutenção e reutilização: Assegurar a preservação dos dados através de ficheiros binários e a criação de representações gráficas através de ficheiros de texto.

2 Análise e Especificação

2.1 Estruturas de Dados Utilizadas

Foram utilizadas listas ligadas para armazenar informações tanto das antenas como dos efeitos nefastos. Esse tipo de estrutura permite manipulação dinâmica dos elementos, facilitando inserções e remoções sem necessidade de alocar um espaço fixo de memória.

2.1.1 Estrutura Antena

A estrutura Antena foi concebida para representar cada antena presente na cidade. Esta estrutura contém informações essenciais, nomeadamente:

- linha e coluna: Coordenadas que definem a posição da antena na matriz que representa o mapa urbano;
- frequência: Caracter que indica a frequência de operação da antena, determinando o sinal emitido;
- id: Identificador único que permite a rápida localização e manipulação da antena sem necessidade de recorrer às coordenadas;
- next: Apontador para a próxima antena na lista, o que permite a constituição de uma lista ligada dinâmica. Este mecanismo facilita operações como inserção, remoção e alteração dos registos;

2.1.2 Estrutura Nefasto

A estrutura Nefasto destina-se a registar as localizações onde ocorre o efeito nefasto. Os campos desta estrutura são:

- linha e coluna: Coordenadas que definem a posição do efeito na matriz;
- id Antena 1 e id Antena 2: Identificadores das duas antenas que, em conjunto, desencadeiam o efeito nefasto;
- next: Apontador para o próximo registo de efeito na lista, possibilitando a criação de uma lista ligada dinâmica para os efeitos. Esta abordagem permite a gestão ordenada e dinâmica dos efeitos calculados;

2.2 Dados a Entrar e Preservação em Ficheiros

Os dados das antenas e dos efeitos nefastos são lidos a partir de ficheiros de texto e armazenados em ficheiros binários, permitindo uma recuperação eficiente.

2.3 Funcionalidades e Requisitos

2.3.1 Funcionalidades Implementadas

O sistema implementa várias funcionalidades que abrangem:

- Gestão de antenas: Criação, inserção ordenada, remoção, alteração e procura de antenas na lista;
- Detecção dos efeitos nefastos: Cálculo dos pontos onde ocorre o efeito nefasto para cada par de antenas com a mesma frequência, utilizando fórmulas que consideram a posição relativa das antenas;
- Preservação e recuperação dos dados: Gravação dos registos das antenas e dos efeitos nefastos em ficheiros binários e geração de uma representação gráfica em ficheiros de texto;

2.3.2 Especificação dos Dados de Entrada e Saída

- Dados de entrada: Ficheiro de texto contendo a representação matricial do mapa urbano, onde cada célula da matriz indica a presença ou ausência de uma antena, bem como a respetiva frequência;
- Dados a preservar: Estruturas dinâmicas (listas ligadas) que armazenam os registos das antenas e dos efeitos nefastos;
- Formato de saída:
 - Ficheiro binário que contém os dados das antenas e dos efeitos, permitindo a reconstrução da lista em execuções futuras;
 - Ficheiro de texto com a representação gráfica do cenário, onde as antenas são identificadas pela sua frequência e os efeitos nefastos são marcados com um símbolo;

3 Implementação em Linguagem C

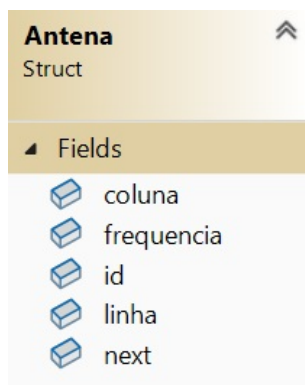
3.1 Estruturas e Definições

3.1.1 Estrutura Antena

A estrutura **Antena** é definida no ficheiro dados.h e contém os campos necessários para a identificação e gestão das antenas. O apontador **next** é fundamental para criar a ligação entre os elementos, constituindo assim uma lista ligada dinâmica.

- **linha** e **coluna**: Coordenadas da antena na matriz;
- **frequência**: Indica a frequência de operação da antena;
- **id**: Identificador único da antena;
- **próximo**: Apontador para a próxima antena na lista ligada.

Representada na imagem a seguir:



3.1.2 Estrutura Nefasto

A estrutura **Nefasto** é utilizada para armazenar informações sobre os efeitos nefastos detetados, incluindo:

- **linha** e **coluna**: Coordenadas do efeito na matriz;
- **idAntena1** e **idAntena2**: Identificadores das antenas responsáveis pelo efeito;
- **próximo**: Apontador para o próximo efeito nefasto na lista ligada.

Representada na imagem a seguir:



3.2 Organização do Código

O código encontra-se organizado em três ficheiros principais, de forma a separar a implementação dos conceitos de interface e de dados:

- **funcoes.c**: Contém a implementação de todas as funções, desde a criação e manipulação das listas de antenas e efeitos, até à gestão dos ficheiros. Este ficheiro é rico em comentários no formato Doxygen, facilitando a compreensão do seu funcionamento;
- **funcoes.h**: Reúne as assinaturas das funções implementadas em **funcoes.c**, servindo de interface para o resto do programa, nomeadamente a função **main**;
- **dados.h**: Define as estruturas de dados (**Antena** e **Nefasto**) e inclui as bibliotecas necessárias, garantindo que os registos são definidos de forma única através do **pragma once**. A diretiva **CRT-SECURE-NO-WARNINGS** é também aplicada para permitir o uso de funções de I/O (neste caso **fopen**) sem restrições;

3.3 Implementação das Funções

3.3.1 Funções de Gestão de Antenas

As funções dedicadas à gestão das antenas incluem:

- **criaAntena**: Aloca memória e inicializa os campos de um novo registo de antena;

- **inserirOrdenado:** Insere o registo da antena na lista de forma ordenada, recorrendo às coordenadas (linha e coluna) para definir a posição correta;
- **removeAntena:** Remove um registo da lista com base nas coordenadas fornecidas, assegurando a correta atualização dos apontadores; **:** Permite a modificação dos dados de uma antena, identificada através do seu id, mantendo a consistência da lista;
- **ProcuraAntena:** Procura e retorna um apontador para uma antena com o id indicado, facilitando a sua manipulação;

3.4 Funções para os Efeitos Nefastos

No que toca à gestão dos efeitos nefastos, destacam-se as seguintes funções:

- **inserirEfeito:** Cria um novo registo para um efeito nefasto e insere-o na lista, mantendo a ordem baseada nas coordenadas. O apontador next é utilizado para ligar cada registo, criando assim uma lista ligada;
- **atualizaEfeito:** Percorre a lista de antenas e, para cada par de antenas com a mesma frequência, calcula os pontos de efeito utilizando fórmulas que determinam as coordenadas dos efeitos. Para cada par, dois registos são gerados e inseridos na lista de efeitos, garantindo que todos os pontos relevantes são registados;
- **mostraLista:** Apresenta, de forma tabular, os registos das listas de antenas e efeitos, permitindo a verificação visual e a depuração do sistema;

3.5 Gestão de Ficheiros

A preservação e recuperação dos dados são alcançadas através de duas abordagens complementares:

- **Gravação em ficheiro binário:**
 - A função `gravarFicheiroB` grava sequencialmente os dados das antenas e dos efeitos num ficheiro binário.
 - Este método permite armazenar os dados de forma compacta e eficiente, sem a necessidade de conversão para formato textual.
- **Geração de ficheiro de texto:**
 - A função `gravarMatrizTxt` gera uma representação gráfica do cenário, onde cada antena é representada pelo seu identificador de frequência e os efeitos nefastos são marcados com um símbolo
 - Esta abordagem facilita a análise visual e a verificação da disposição das antenas e dos pontos de efeito no mapa.
- **Leitura de ficheiro binário:**
 - A função `lerFicheirobinario` permite reconstituir a lista de antenas a partir dos dados armazenados, possibilitando a recuperação e continuidade do trabalho em execuções subsequentes.

3.6 Cálculos e Lógica dos Efeitos Nefastos

A lógica para determinar os efeitos nefastos é implementada na função `atualizaEfeito`. Para cada par de antenas com a mesma frequência, os seguintes cálculos são efetuados:

- Cálculo do primeiro ponto de efeito:
 - Fórmula: $\text{efeitoX1} = 2 * \text{linha}(\text{antena } 1) - \text{linha}(\text{antena } 2)$;
 - Fórmula: $\text{efeitoY1} = 2 * \text{coluna}(\text{antena } 1) - \text{coluna}(\text{antena } 2)$
- Cálculo do segundo ponto de efeito:
 - Fórmula: $\text{efeitoX2} = 2 * \text{linha}(\text{antena } 2) - \text{linha}(\text{antena } 1)$
 - Fórmula: $\text{efeitoY2} = 2 * \text{coluna}(\text{antena } 2) - \text{coluna}(\text{antena } 1)$

Estes pontos são posteriormente inseridos na lista de efeitos através da função `inserirEfeito`, que garante a ordenação e a preservação dos registros.

4 Conclusão

O presente projeto demonstrou a implementação de um sistema robusto para a gestão de antenas e dos respectivos efeitos nefastos, consolidando os conhecimentos adquiridos na disciplina de Estruturas de Dados Avançadas. A utilização de estruturas de dados dinâmicas, notadamente listas ligadas, permitiu uma gestão eficiente e flexível dos registros, tanto para as antenas como para os efeitos. A modularização do código e a clara separação entre interface e implementação, juntamente com a documentação produzida com Doxygen, contribuem para a manutenção e compreensão do sistema, independentemente do nível de conhecimento do utilizador em linguagem C.

A preservação dos dados através de ficheiros binários e a geração de representações gráficas em ficheiros de texto reforçam a aplicabilidade do sistema em contextos reais, demonstrando a sua utilidade e a relevância dos conceitos de gestão de memória e de estruturas dinâmicas. Por fim, o projeto representa uma base sólida para futuras extensões e melhorias, servindo como referência para a implementação de soluções de dimensão média na área das Estruturas de Dados Avançadas.

No futuro, pretende-se expandir este trabalho na segunda fase do projeto, onde serão aplicados conceitos avançados de teoria dos grafos e programação em C para resolver o mesmo problema computacional. Essa evolução permitirá uma análise mais aprofundada da conectividade entre as antenas e a otimização da sua disposição para minimizar interferências e maximizar a eficiência da comunicação.

5 Referências

- Leitura e escrita de estruturas em ficheiros em C - GeeksforGeeks
- Função `fwrite()` em C - GeeksforGeeks
- Função `fread()` em C - GeeksforGeeks
- Listas Ligadas em C - GeeksforGeeks
- Alocação Dinâmica de Memória em C - GeeksforGeeks
- Função `fopen()` em C - GeeksforGeeks