**Escola Superior de Tecnologia e Gestão**

**Licenciatura em Engenharia Informática**

**Estruturas de Dados**

**Ano Letivo 2023/24**

**Trabalhos Final**

**Elaborado em: 2024/06/02**

**Guilherme Rodrigues a2020154390**

**Índice**

[List of Figures ii](#__RefHeading___Toc974_2925045303)

[1 Introdução 1](#__RefHeading___Toc976_2925045303)

[2 Explicação do Código 2](#__RefHeading___Toc978_2925045303)

[2.1 Funcionarios 4](#__RefHeading___Toc980_2925045303)

[2.1.1 Registo de Funcionarios 4](#__RefHeading___Toc982_2925045303)

[2.1.2 Internetwork Operating System (IOS) 4](#__RefHeading___Toc984_2925045303)

[2.2 Basic Switch and End Device Configuration 4](#__RefHeading___Toc986_2925045303)

[2.3 Protocols and Models 4](#__RefHeading___Toc988_2925045303)

[2.4 Physical Layer 4](#__RefHeading___Toc990_2925045303)

[2.5 Number Systems 4](#__RefHeading___Toc992_2925045303)

[2.6 Data Link Layer 4](#__RefHeading___Toc994_2925045303)

[3 Conclusion 5](#__RefHeading___Toc996_2925045303)

[4 References 6](#__RefHeading___Toc998_2925045303)

# List of Figures

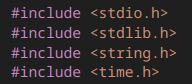
# Introdução

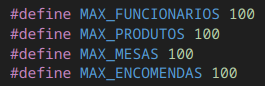
O objetivo é realizar um sistema em C para gerenciar o funcionamento de um café, desde a organização dos funcionários e produtos até o registro de vendas e geração de relatórios diários. O objetivo principal é garantir a eficiência operacional, o controlo do stock e a satisfação do cliente.

# Explicação do Código

Antes de começar qualquer função foram importadas 4 bibliotecas sendo elas “stdio.h”, “stdlib.h”, “string.h” e “time.h”. Esta ultima nao foi utilizada em definitivo mas foi deixado uma função como comentario para demonstar como a utilizariamos para usar o dia atual do sistema para dar nome a um ficheiro.

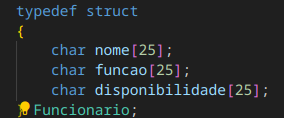
Em seguida definimos tabem o tamanho maximo de cada tabela que vamos utilizar neste projeto. Como serão utilizadas 4 tabelas foram derinidos 4 tamanhos maximos que depois de uma análise tardia se chegou a conclusão de que talvez poderia ter definido apenas uma variavel constante.  
  
Em seguuda criou-se as 6 estruturas necessárias. A estrutura “Funcionario”, a estrutura “Produto”, a estrutura “Encomenda”, a estrutura “IntemConta”, a estrutura “Pedido” e por fim a estrutura “Mesa”.

Figure 1: Bibliotecas utilizadas

Figure 2: Definição das constantes

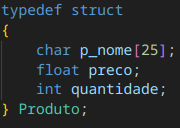
* Etrutura Funcionário

A estrutura “Funcionario” que irá guardar a infromação dos funcionarios. Esta estrutura guarda 3 tipos de dados, “char nome[25]”, “char funcao[25]” e “char disponibilidade[25]”. Todas possuem um tamanho fixo de 25, o ue significa que so podemos guardar até 24 caracteres. Cada uma destas varaveis irá guardar informação referente ao nome da mesma.

Figure 3: Estrutura Funcionario

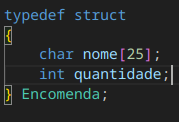
* Estrutura Produto

Esta estrutura tambem guarda 3 variaveis, “char p\_nome[25]”, “float preco” e “int quantidade”. A semelhança da anterior e das restantes estruturas o nome das variáveis guardam informação referente ao nome da variavél.

Figure 4: Estrutura Produto

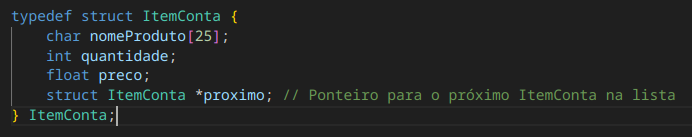
* Estrutura Encomenda

Esta estrutura guarda 2 variaveis, “char nome[25]” e “int quantidade”. É pretendido que a primeira variável guarde o nome do produto que se pretende ser encomendado e não que seja literalmetne o nome da encomenda.

Figure 5: Estrutura Encomenda

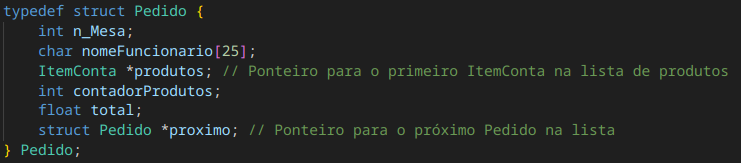
* Estrutura Item Conta

Esta estrutura para alem de 3 variáveis “char nomeProduto[25]”, “int quantidade” e “float preco” tambem possui um ponteiro para ela própria para que seja criado uma lista ligada “struct ItemConta \*proximo” abrindo assim a possibilidade de guardar um número dinamico de intens sem precisar alocar previamente um grande bloco de memória.

Figure 6: Estrutura ItemConta

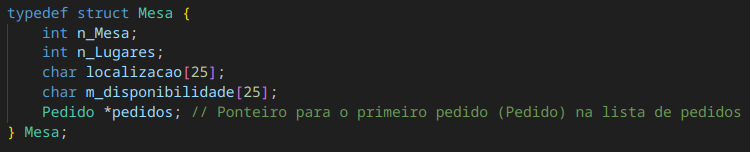
* Estrutura Pedido

A estrutura Pedido possui dois ponteiros, o primeiro “ItemConta \*produtos” aponta para o primeiro ItemConta na lista de produtos e o segundo aponta para o proximo pedido. Assim temos uma lista de pedidos e cada pedido possui uma lista de produtos. O resto das variaveis são “int n\_Mesa”, “char nomeFuncionario”, “int contadorProdutos” e “float total”.

Figure 7: Estrutura Pedido

* Estrutura Mesa

Por fim a estrutura Mesa tem um ponteiro que aponta para o primeiro pedido na lista de pedidos fazendo com que cada mesa tenha uma lista de pedidos. O restante da variaveis são “int n\_Mesa”, “int n\_Lugares”, “char localizacao[25]” e “char m\_disponibilidade”.

Figure 8: Estrutura Mesa

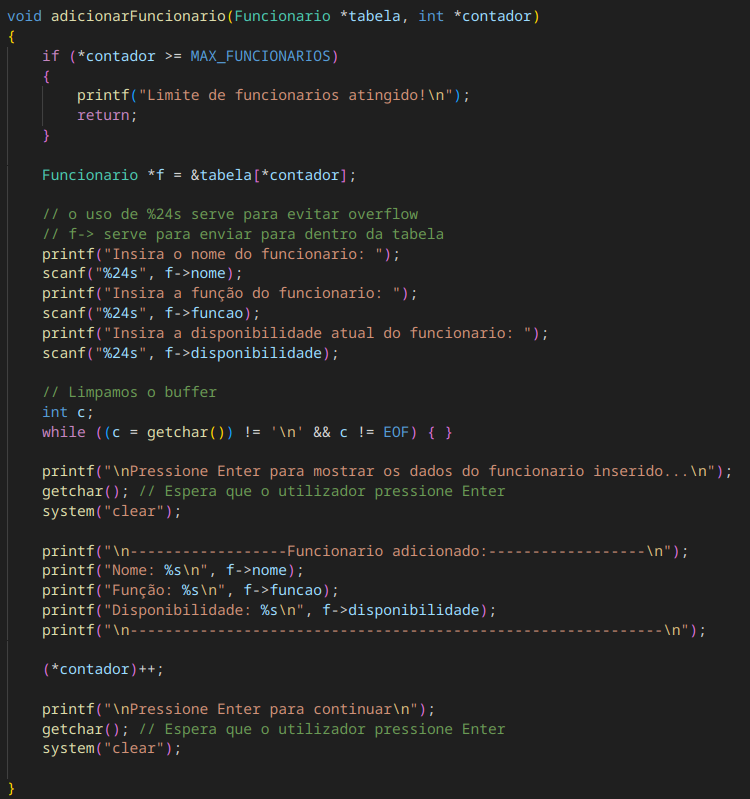
## Funcionarios

### Registo de Funcionarios

A afunçao “adicionarFuncionario” recebe 2 parametros “Funcionario \*tabela” e “int \*contador”. Estes parametros são: um ponteiro para a tabela funcionarios e um ponteiro que armazena o numero de funcionarios registados.

A primeira verificação que esta função faz é verificar se o número de funcionários (\*contador) já atingiu o limite máximo (MAX\_FUNCIONARIOS), caso o limite tenha sido atingido, imprime uma mensagem de erro e retorna da função.

Aloca uma variável e do tipo Funcionario e inicializa-a com o endereço do próximo elemento disponível na tabela (&tabela[\*contador]).



### Internetwork Operating System (IOS)

xxx

## Basic Switch and End Device Configuration

## Protocols and Models

## Physical Layer

## Number Systems

## Data Link Layer

xxx

# Conclusion

# References

References serve three primary purposes—documentation, acknowledgment, and directing or linking the reader to additional resources. The author may cite a reference to support their own arguments or lay the foundation for their theses (documentation); as a credit to the work of other authors (acknowledgment); or to direct the reader to more detail or additional resources (directing or linking). References are a critical element of a manuscript.

To manage the references, you must use a reference manager (from the built in Microsoft Source Manager to more capable and platform-independent applications such as Zotero).

There are several conventions to present references, with APA (American Psychological Association), ACM (Association for Computing Machinery), and IEEE (Institute of Electrical and Electronics Engineers) being the most

To satisfy your curiosity for the time being, you may find hints on the ACM requirements at <http://www.acm.org/pubs/submissions/submission.htm> and look up examples of the differences between the conventions of the ACM, APA, and IEEE at <http://www.library.dal.ca/subjects/csci_ref.htm>. Bellow, you are given a few examples of citations complying with the requirements of the ACM

An article in a journal:

ABDELBAR, A.M., AND HEDETNIEMI, S.M. 1998. Approximating MAPs for belief networks in NP-hard and other theorems. Artificial Intelligence 102, 21-38.

A book:

GINSBERG, M. 1987. Readings in Nonmonotonic Reasoning. Morgan Kaufmann, Los Altos, CA.

A chapter in a book:

GREINER, R. 1999. Explanation-based learning. In The Encyclopedia of Cognitive Science, R. WILSON AND F. KEIL, Eds. MIT Press, Cambridge, MA, 301-303.

An article in conference proceedings:

MAREK, W., AND TRUSZCZYNSKI, M. 1989. Relating autoepistemic and default logics. In Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning, Toronto, Canada, May 1989, H. BRACHMAN AND R. REITER, Eds. Morgan Kaufmann, San Mateo, CA, 276-288.