

**Instituto Politécnico do Cávado e do Ave**

Flexible Job Shop

Parte I

**Diogo Alexandre Mendes Simões**

**Projecto de Licenciatura EIM**

**em Estruturas de Dados Avançadas**

Orientador:

Docente João Carlos Silva, IPCA- Instituto Politécnico do Cávado e do Ave

Abril 2022



Flexible Job Shop, C

Diogo Alexandre Mendes Simões

# RESUMO

Este trabalho, teve como objetivo o desenvolvimento de aplicação em linguagem C, utilizando estruturas de dados dinâmicas.

O trabalho foi dividido em duas partes, que correspondentes a diferentes etapas: Definição de estrutura de um job e operações, Definição de um conjunto finito de jobs e inserção dos mesmos.

Na aplicação desenvolvida na primeira parte, o objetivo é definir um Job e a realização de operações, tendo aplicado listas ligadas multidimensionais desenvolvido em linguagem de programação C.

Palavras-Chave: Estruturas de dados; Linguagem C; Json; Listas ligadas

# ABSTRACT

This work aimed to develop an application in C language, using dynamic data structures.

The work was divided into two parts, corresponding to different stages: Definition of a job's structure and operations, Definition of a finite set of jobs and their insertion.

In the application developed in the first part, the objective is to define a Job and the performance of operations, having applied multidimensional linked lists developed in C programming language.

Keywords: Data structures; C language; Json; linked lists

Índice

[RESUMO 3](#_Toc92145854)

[ABSTRACT 4](#_Toc92145855)

[Introdução 9](#_Toc92145856)

[ Objetivos 10](#_Toc92145857)

[ Justificativa 10](#_Toc92145858)

[ Motivação 10](#_Toc92145859)

[System IHosp 11](#_Toc92145860)

[ANÁLISE E ESPECIFICAÇÃO DO SISTEMA 12](#_Toc92145861)

[Pormenorização do produto 13](#_Toc92145862)

[ Área de Utente 13](#_Toc92145863)

[ Área do Médico 13](#_Toc92145864)

[ Área do Administrador 14](#_Toc92145865)

[Funcionamento do Sistema IHosp 15](#_Toc92145866)

[ Funcionamento da Área de Utente 15](#_Toc92145867)

[Sistema IHosp de Informação 16](#_Toc92145869)

[ Funcionamento da Área dos Profissionais de Saúde 17](#_Toc92145870)

[Sistema IHosp de Gestão de Departamentos 18](#_Toc92145871)

[ Funcionamento da Área de Administrativa 19](#_Toc92145872)

[Sistema IHosp de Gestão Administrativa 20](#_Toc92145873)

[Sistema IHosp de Gestão Financeira 24](#_Toc92145874)

[Cronograma Global de fases e etapas do programa 26](#_Toc92145875)

[Proposta financeira que discrimine os custos associados à implementação da solução 26](#_Toc92145877)

[Discriminação e valorização dos proveitos 27](#_Toc92145879)

[Conclusão 28](#_Toc92145880)

Índice de Figuras

[Figura 1-Logotipo 9](file:///C:\Users\diogo\OneDrive\Ambiente%20de%20Trabalho\Gestão%20Saude\Gestao.docx#_Toc92145882)

[Figura 2-Área Utente 11](#_Toc92145883)

[Figura 3-Área Médica 11](#_Toc92145884)

[Figura 4-Área Financeira 12](#_Toc92145885)

[Figura 5-Área Administrativa 12](#_Toc92145886)

[Figura 6-Página Inicial Hospital 13](#_Toc92145887)

[Figura 7-Área de Utente IHosp 14](file:///C:\Users\diogo\OneDrive\Ambiente%20de%20Trabalho\Gestão%20Saude\Gestao.docx#_Toc92145888)

[Figura 8-Página Inicial Hospital 15](#_Toc92145889)

[Figura 9-Área Profissional Saúde 15](#_Toc92145890)

[Figura 10-Área do Profissionais Saúde Software 16](#_Toc92145891)

[Figura 11-Página Inicial Hospital 17](#_Toc92145892)

[Figura 12- Área do Administrador Software 17](#_Toc92145893)

[Figura 13-Área Profissionais Saúde 18](#_Toc92145894)

[Figura 14-Área de gestão profissionais 18](#_Toc92145895)

[Figura 15-Área de Vizualização de Altas Hospitalares 19](#_Toc92145896)

[Figura 16-Área Gestão Camas Hospitalares 20](#_Toc92145897)

[Figura 17- Área do Administrador Software 22](#_Toc92145898)

[Figura 18-Área Financeira IHosp 22](#_Toc92145899)

[Figura 19- Cronograma Global 24](#_Toc92145900)

[Figura 20-Proposta Financeira 24](#_Toc92145901)

# 

# Introdução

No âmbito da unidade curricular de Estruturas de dados avançadas, do segundo semestre do primeiro ano do curso de Engenharia Informática Médica do Instituto Politécnico do Cavado e do Ave, no ano letivo 2016/2017 foi-nos proposto a elaboração de um trabalho prático que está dividido em duas partes.

Este trabalho surge no contexto da avaliação contínua composta por um trabalho prático.

O objetivo principal deste trabalho é levar ao melhor entendimento dos conhecimentos adquiridos ao longo do semestre e a melhor aplicação dos mesmos.

Assim, com este trabalho pretendemos sedimentar mais os nossos conhecimentos sobre os temas abordados nas respetivas aulas.

O trabalho realizado individual tem como objetivo a elaboração de um programa em c para o problema de escalonamento denominado Flexible Job Shop Problem (FJSSP).

## Objetivos

Espera-se, com a execução desse projeto, sedimentar os conhecimentos relativos a definição e manipulação de estruturas de dados dinâmicos em linguagem de programação C no desenvolvimento de uma solução digital para o problema de escalonamento denominado por Flexible Job Shop Problem, permitindo gerar uma proposta de escalonamento para a produção d um produto envolvendo várias operações e a utilização de várias máquinas.

## Justificativa / Motivação

A intensa disputa por novos mercados e o apogeu das novas tecnologias tem obrigado as empresas a desenvolverem constantes estratégias que possam dinamizar seus processos internos e inovar os seus sistemas.

O Software a ser desenvolvida visa prestar suporte as organizações focadas no processo são oficinas de artesanato, restaurantes, empresas de mecânica, oficinas de pintura, gráficas comerciais e outras indústrias que fabricam produtos personalizados em pequenos lotes.

.

**Contexto**

Em este projeto que nos foi proposto, que está dividido em duas partes, iremos abordar em este relatório apenas a primeira parte no projeto.

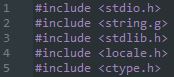
Em esta parte do projeto será necessario realizar os seguintes requisitos:

* Definição de uma estrutura de dados dinâmica para a representação de um job com um conjunto finito de n operações;
* Armazenamento/leitura de ficheiro de texto com representação de um job;
* Inserção de uma nova operação;
* Remoção de uma determinada operação;
* Alteração de uma determinada operação;
* Determinação da quantidade mínima de unidades de tempo necessárias para completar o job e listagem das respetivas operações;
* Determinação da quantidade máxima de unidades de tempo necessárias para completar o job e listagem das respetivas operações;
* Determinação da quantidade média de unidades de tempo necessárias para completar uma operação, considerando todas as alternativas possíveis;

# Descrição da implementação

Para desenvolver o programa e para que este funcione corretamente, foi necessário que todos os procedimentos e funções criados interagissem de forma eficaz, sem erros e evitando situações do tipo “beco sem saída” que ocorrem quando as aplicações bloqueiam ou ficam presas em estados sem retorno. O resultado final é o código fonte que se encontra anexado a este relatório, sendo que nesta seção serão explicadas algumas técnicas implementadas.

## Bibliotecas

****

Estas instruções permitem incluir bibliotecas ao programa. Além das necessidades normais de input/output (stdio.h), alocação de memória (stdlib.h), manipulação de strings (string.h) e operações com diferentes tipos de variáveis (ctype.h), foi necessário recorrer à biblioteca locale.h para poder incluir carateres especiais em língua portuguesa, que serão largamente utilizados ao longo do programa.

## Descrição Técnica

Para a realização deste trabalho foi necessário recorrer à criação de diversas estruturas.

Uma estrutura é um conjunto de variáveis de tipos distintos ou não, agrupadas sob um único nome. As variáveis que compõem a estrutura são chamadas membros, campos ou elementos.

Com a palavra-chave **struct** declara-se um novo tipo de dado.

A palavra seguinte será o seu identificador ou tipo de estrutura.

Quando se faz a declaração **struct data:**

**- struct** é um tipo distinto de dado como int ou float.

- **data** é referência a um tipo específico de estrutura.

Os membros são declarados entre chaves. A declaração termina com um “ponto e vírgula” (;).

Foram criadas as seguintes estruturas:

**Funções**

Na linguagem C, a função é um conjunto de comandos que realiza uma tarefa especifica em um módulo dependente de código. A função é referenciada pelo programa principal através do nome atribuído a ela.

A utilização de funções visa modularizar um programa em várias partes, no qual é muito comum em programação estruturada. Desta forma podemos dividir um programa em várias partes, no qual cada função realiza uma tarefa bem definida.

No nosso programa estão implementadas as seguintes funções:

**Inserir uma nova Operação**

**Remover uma determinada operação**

**Alteração de uma determinada operação**

**Determinar a quantidade mínima de unidades de tempo necessárias para completar o job e listagem das respetivas operações**

# 

# Conclusão

Após terminarmos a realização do trabalho, conseguimos por em prática os conhecimentos dados nas aulas, embora nos tenhamos deparado com bastantes dificuldades.

Apesar de todas as dificuldades encontradas neste trabalho, conseguimos superar algumas com sucesso.

Este trabalho consegue-nos demonstrar um pouco o nivel de dificuldade que podemos ter no exterior a nivel profissional.

Contudo, este trabalho vem nos trazer mais experiência e conhecimento da disciplina , preparando-nos assim para a realização de mais trabalhos no futuro com outros programadores/empresas.

# Bibliografia

Na realização deste trabalho, fomos encontrando várias dificuldades ao longo do projeto, e ultrapassadas com sucesso, como foi dito anteriormente.

Usamos assim livros de programação, os conteúdos e códigos usados nas aulas da Unidade Curricular Estrutura de dados Avançadas , e também ajuda de familiares para a execução deste trabalho.