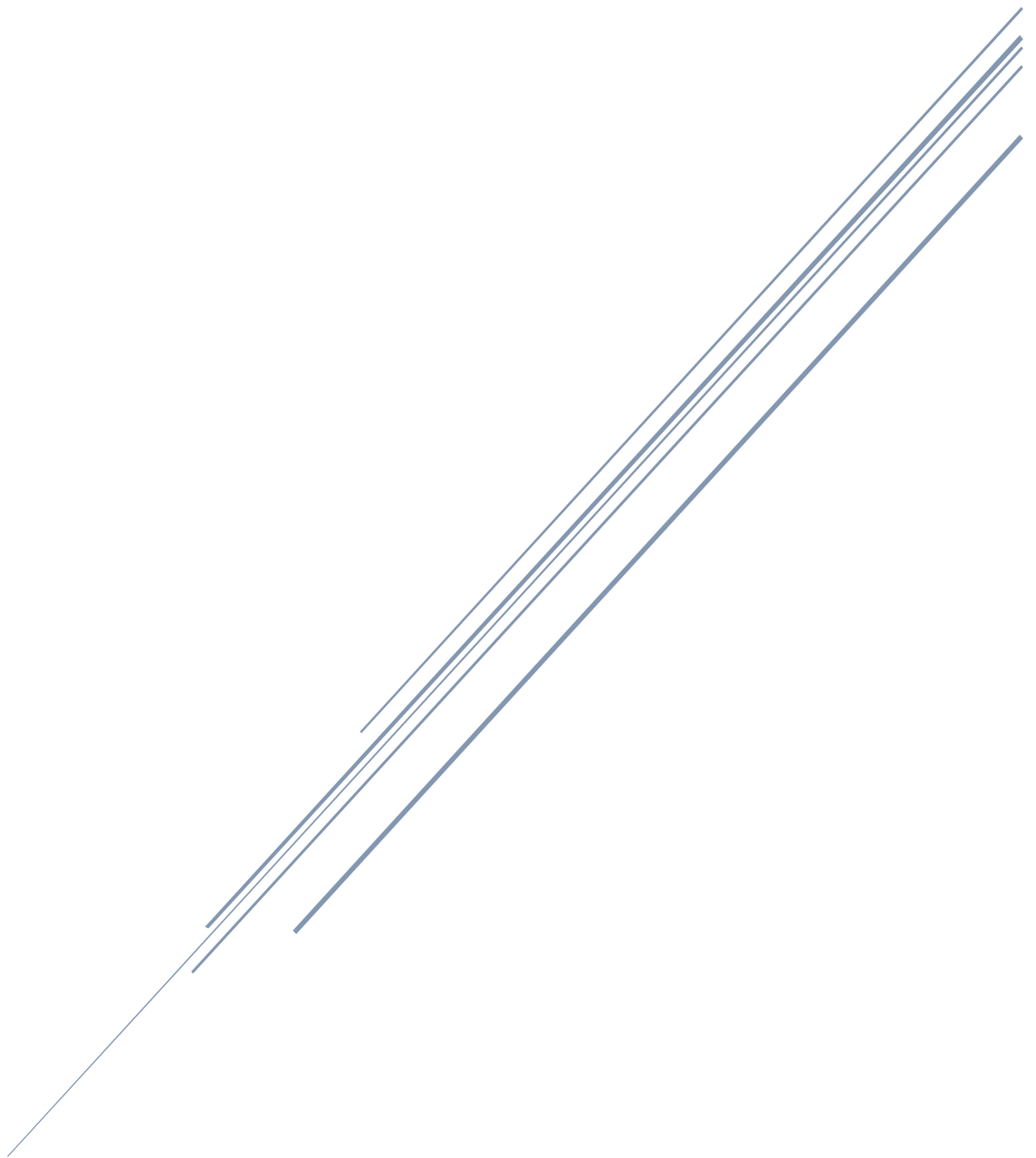


# ESTRUTURA DE DADOS AVANÇADAS

Relatório da fase 1 do Projeto



Rui Pinto - 18450  
Engenharia de Sistemas Informáticos

## Índice

Introdução .....	2
Propósitos e Objetivos .....	3
Estrutura de Dados .....	4
Testes realizados.....	6
Conclusão .....	7

## Introdução

O trabalho desenvolvido no âmbito da unidade curricular “Estrutura de Dados Avançadas”, tem como objetivo desenvolver um programa que permita gerir a produção de um produto envolvendo várias operações e a utilização de várias máquinas.

Este projeto tem ainda a finalidade de aplicar conteúdos lecionados anteriormente da linguagem C, aprofundando-os, e ainda aplicar novos conteúdos como algoritmos de procura e de ordenação, listas dinâmicas e apontadores.

## Propósitos e Objetivos

Tal como foi referido, o objetivo do programa é gerir a produção de um determinado produto que é tratado como um Job, com as várias operações necessárias e as máquinas que serão utilizadas.

Na fase 1 do projeto foi-nos proposto gerir a produção de apenas um produto, podendo adicionar, editar ou remover as operações e determinar o tempo máximo, mínimo e médio necessário para completar o Job, apresentando as respetivas operações.

O programa também terá de abrir o ficheiro de texto que terá guardado toda a informação necessária para depois caso seja preciso modificar alguma informação ou até mesmo adicionar operações. No fim a informação será guardada no mesmo ficheiro.

## Estrutura de Dados

A estrutura do projeto é organizada por três ficheiros:

- **Main.c:** Este é o programa principal porque é onde corre o programa. É constituído pelas bibliotecas necessárias para o funcionamento do programa e pela função main.
- **Header.h:** Esta é a biblioteca que foi criada para colocar todas as funções criadas outros ficheiros e serem utilizadas no main.c, e também se encontra as estruturas de dados.
- **Functions.c:** Neste ficheiro encontra-se todas as funções utilizadas no programa final.



De modo a guardar informações importante para a execução do trabalho foram utilizadas as seguintes estruturas:

- **Operation:** A seguinte estrutura vai ser a mais importante do trabalho, uma vez que irá guardar informações referentes a cada operação. As operações são identificadas com um número de identificação, a quantidade de máquinas utilizadas para executar a operação, uma variável inteira para definir o tempo para cada máquina e um apontador para a próxima operação;

```
typedef struct operation {  
    int id;  
    int quantMachines;  
    int *machineAndTime;  
    struct operation *next;  
}operation;
```

- **Job:** A estrutura contém apenas um apontador para que quando uma operação foi removida o seu número de identificação seja ocupado pelo seguinte.

```
typedef struct job {  
    operation *op;  
}job;
```

As funções utilizadas no programa encontram-se no ficheiro functions.c e estão na biblioteca header.h para que sejam utilizadas no main.c.

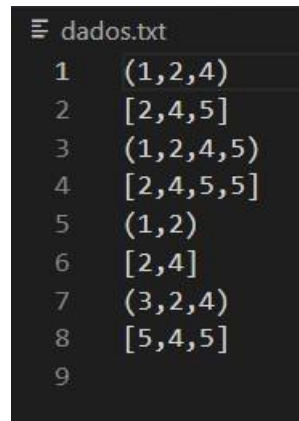
```
job *inicializarJob();  
operation *inicializarOperation();  
void verificarDadosNoFicheiro(operation *op, int *idCont);  
void guardarDadosNoFicheiro(operation *op);  
void menu(int *opcao);  
void insertNewOperation(operation *op, int *idCont);  
void listOperation(operation *op);  
void removeOperation(job **jobList);  
void editOperation(operation *op);  
void determineShortestTime(operation *op);  
void determineLongestTime(operation *op);
```

## Testes realizados

Durante o desenvolvimento do código foram feitos testes num ficheiro de texto. O começo dos testes foi adicionando primeiro as operações que seriam necessárias e de seguida as máquinas com o seu tempo.

Também foram feitos testes para as funções que determinam o máximo e o mínimo de tempo necessário para completar um Job, e por fim toda a informação é guardada no ficheiro de texto chamado dados.txt.

Ficheiro de texto utilizado para a realização dos testes:



```
≡ dados.txt
1 (1,2,4)
2 [2,4,5]
3 (1,2,4,5)
4 [2,4,5,5]
5 (1,2)
6 [2,4]
7 (3,2,4)
8 [5,4,5]
9
```

## Conclusão

Podemos assim dar a fase 1 concluída com sucesso, onde atingimos todos os objetivos propostos. Consideramos um projeto interessante e desafiante, uma vez que conseguimos aperfeiçoar as capacidades desenvolvidas nas aulas, mas também aprender novas técnicas de programação em C, desenvolvendo o nosso conhecimento da linguagem para além do que nos é lecionado.