Estruturas de Dados Avançados

Uma imagem com texto

Descrição gerada automaticamente

Flexible Job Shop Problem

Professor:

João Carlos Silva

Luís Gonzaga Martins Ferreira

Alunos:

Bruno Dantas Aurélio Coelho Dantas: a20807

Maio 31, 2022

LESI

# Índice

[Introdução 4](#_Toc101183262)

[Propósitos e Objetivos 5](#_Toc101183263)

[Estruturas de dados 6](#_Toc101183264)

[Testes 8](#_Toc101183265)

[Conclusão 14](#_Toc101183266)

[Bibliografia 15](#_Toc101183267)

# Índice de Figuras

[Figura 1 Estrutura de dados -> Operations 6](#_Toc101183157)

[Figura 2 Estrutura de dados -> OperationsLst 6](#_Toc101183158)

[Figura 3 Estrutura de dados -> SubOperation 7](#_Toc101183159)

[Figura 4 Testes -> Menu 9](#_Toc101183160)

[Figura 5 Testes -> Leitura de um ficheiro 9](#_Toc101183161)

[Figura 6 Testes -> Escrita de um ficheiro 10](#_Toc101183162)

[Figura 7 Testes -> Adicionar nova operação 10](#_Toc101183163)

[Figura 8 Testes -> Adicionar novos elementos a operação caso já exita na lista 11](#_Toc101183164)

[Figura 9 Testes -> Apagar uma operação da lista 11](#_Toc101183165)

[Figura 10 Testes -> Alterar uma operação 12](#_Toc101183166)

[Figura 11 Testes -> Tempo mínio possível para completar uma operação 12](#_Toc101183167)

[Figura 12 Testes -> Tempo máximo possível para completar uma operação 13](#_Toc101183168)

[Figura 13 Testes -> Tempo medio possível para completar uma operação 13](#_Toc101183169)

# Introdução

O presente relatório pretende documentar o trabalho pratico da disciplina de Estruturas de Dados Avançados.

O trabalho pratico consiste na geração de uma proposta de escalonamento para a produção de e um produto envolvendo várias operações de várias máquinas, minimizando o tempo para sua produção. Este trabalho foi devido em 2 fases, nesta primeira fase consiste na criação e manipulação da parte da Operação.

# Propósitos e Objetivos

O propósito da primeira fase deste trabalho pratico, é a criação de um programa que consiga lidar com várias **Operations** que cada terá um conjunto de **Máquinas** com determinado tempo para terminar aquela **Operation**.

O propósito da segunda fase deste trabalho e complementar a primeira fase com uma estrutura que ira guardar os dados e de uma process plan em que consiste em vários jobs com varias operations que tem varias maquinas e tempos em que podem ser realizados, e na criação de um algoritmo que proponha uma tabela de escalonamento para esse process plan.

# Estruturas de dados

Para elaboração deste trabalho pratico na primeira fase foram criadas 3 esturras em que 2 delas são Queue usando listas duplamente ligadas, uma estrutura auxiliar.

* Esta estrutura ( **\_Operations** ) e uma auxiliar a ( **\_OperationsLst** ), tem como função de guardar o apontador para o primeiro e ultimo elemento da Lista ( **\_OperationsLst** )

Uma imagem com texto

Descrição gerada automaticamente

Figura 1 Estrutura de dados -> Operations

* Na estrutura ( **\_OperationsLst** ) serão guardados todas os identificadores de operação tendo atenção que este identificadores são únicos, também guardara dois apontadores para o primeiro e ultimo elemento da estrutura ( **\_SubOperations** ).

Uma imagem com texto

Descrição gerada automaticamente

Figura 2 Estrutura de dados -> OperationsLst

* Na estrutura ( **\_SubOperations** ) serão guardados os dados das alternativas para a realização de uma operação e serão guardados de forma crescente relativamente a tempo que demora a realização de uma operação.

Uma imagem com texto

Descrição gerada automaticamente

Figura 3 Estrutura de dados -> SubOperation

Para complementar e realizar a segunda fase foram adicionadas mais duas estruturas em que uma e auxiliar para a criação do escalonamento.

* Na estrutura **(\_ProcessPlan)** serão guardados todos o process plan que foram inseridos pelo utilizador. O tipo de lista utilizada para esta estrutura foi um hashtable de listas ligadas em que cada posição pode n elementos.

Uma imagem com texto

Descrição gerada automaticamente

Figura 4 Estrutura de dados -> ProcessPlan

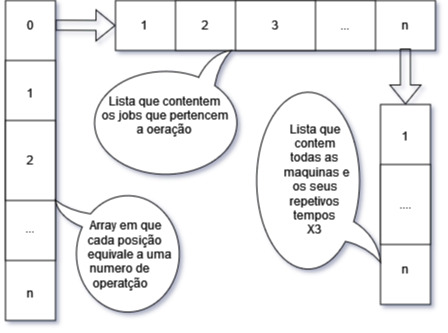
* Na estrutura **( \_MapOperations )** serão guardados os dados do escalonamento por operations

Uma imagem com texto

Descrição gerada automaticamente

Figura 5 Estrutura de dados -> MapOperations

* JobID = Identificar do trabalho
* OperationID = identificador do número da operation
* tested = variável axuliar para reduzir o tempo de iterações
* first = Apontador que aponta para o primeiro elemento da lista Operations
  + usada para reiniciar o position caso já tenha sido percorrido ate ao fim
* position = Apontador que aponta para a lista Operations
  + esta variável e utilizada para percorrer a lista e procurar a maquina que pode ser utilizada neste process plan
* selected = apontador para o valores da Operations selecionados pela função.
* Next = próximo job a ser testado dentro desta opration



# Testes

Foram realizados dois testes a programa um com dois ficheiros diferentes.

1. Ficheiro **A “job.txt”**
   1. **Um simples ficheiro que foi criado para auxiliar na criação do programa-**
   2. **Tem 3 diferentes operações cada operação tem entre 2 a 4 alternativas de escolha de tempo**

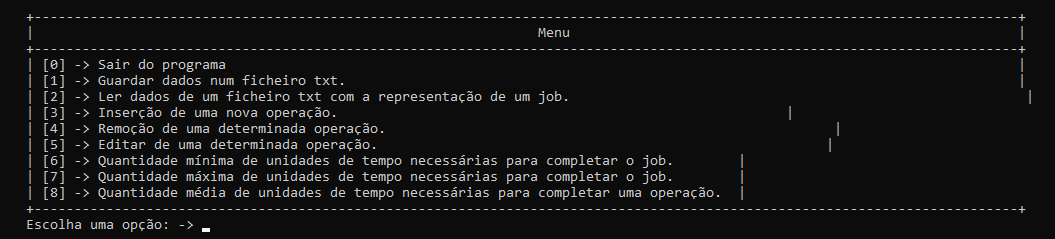


Figura 6 Testes -> Menu

Uma imagem com texto

Descrição gerada automaticamente

Figura 7 Testes -> Leitura de um ficheiro

Uma imagem com texto

Descrição gerada automaticamente

Figura 8 Testes -> Escrita de um ficheiro

Uma imagem com texto

Descrição gerada automaticamente

Figura 9 Testes -> Adicionar nova operação

Uma imagem com texto

Descrição gerada automaticamente

Figura 10 Testes -> Adicionar novos elementos a operação caso já exita na lista

Uma imagem com texto

Descrição gerada automaticamente

Figura 11 Testes -> Apagar uma operação da lista

Uma imagem com texto

Descrição gerada automaticamente

Figura 12 Testes -> Alterar uma operação

Uma imagem com texto

Descrição gerada automaticamente

Figura 13 Testes -> Tempo mínio possível para completar uma operação

Uma imagem com texto

Descrição gerada automaticamente

Figura 14 Testes -> Tempo máximo possível para completar uma operação

Uma imagem com texto

Descrição gerada automaticamente

Figura 15 Testes -> Tempo medio possível para completar uma operação

# Conclusão

Neste trabalho foi abordado as definições e manipulações de estruturas de dados dinâmicas na linguagem de programação **C**.

Foram compridos todos os objetivos propostos no enunciado.

Este trabalho foi muito importante para meu a aprofundamento de Estruturas de dados dinâmicas uma vez que elas são a base de muitas das programações mais atuais e também me permitiu aperfeiçoar a minhas competências de investigação.

# Bibliografia

* MakeFile
  + <https://www.embarcados.com.br/introducao-ao-makefile/>
  + <https://makefiletutorial.com/>
* Linguagem C
  + <https://www.cplusplus.com/>
  + <https://pt.stackoverflow.com/>
  + Moodle da disciplina.