

Projeto de EDA (Fase 1)

Generated by Doxygen 1.9.3

1 Data Structure Index	1
1.1 Data Structures	1
2 File Index	3
2.1 File List	3
3 Data Structure Documentation	5
3.1 _listMachine Struct Reference	5
3.1.1 Detailed Description	5
3.2 _listMacOp Struct Reference	5
3.2.1 Detailed Description	6
3.3 _listOperation Struct Reference	6
3.3.1 Detailed Description	6
4 File Documentation	7
4.1 functions.c File Reference	7
4.1.1 Detailed Description	8
4.1.2 Function Documentation	9
4.1.2.1 avg_time()	9
4.1.2.2 change_operation()	9
4.1.2.3 exist_mac()	9
4.1.2.4 exist_MacOp()	10
4.1.2.5 exist_op()	10
4.1.2.6 freeMachines()	11
4.1.2.7 freeMacOp()	11
4.1.2.8 freeOperations()	11
4.1.2.9 get_max_time()	11
4.1.2.10 head_insert_machine()	12
4.1.2.11 head_insert_macop()	12
4.1.2.12 head_insert_operation()	13
4.1.2.13 insert_operation()	13
4.1.2.14 max_time()	13
4.1.2.15 menu_job()	14
4.1.2.16 min_time()	14
4.1.2.17 readMachines()	14
4.1.2.18 readMacOp()	15
4.1.2.19 readOperations()	15
4.1.2.20 remove_MacOp()	15
4.1.2.21 remove_operation()	16
4.1.2.22 showMachineList()	16
4.1.2.23 showMacOpList()	16
4.1.2.24 showOperationList()	17
4.1.2.25 writeMachines()	17

4.1.2.26 writeMacOp()	17
4.1.2.27 writeOperations()	18
4.2 functions.h	18
4.3 main.c File Reference	18
4.3.1 Detailed Description	19
4.3.2 Function Documentation	19
4.3.2.1 main()	19
4.4 structs.h File Reference	19
4.4.1 Detailed Description	20
4.5 structs.h	20
Index	21

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

_listMachine	Estrutura que representa a lista das máquinas	5
_listMacOp	Estrutura intermedia entre a lista das maquinas e das operações	5
_listOperation	Estrutura que representa a lista das operações	6

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

functions.c		
	Funções do programa	7
functions.h	??
main.c		
	Solução para problema de escalonamento (Flexifile job shop problem)	18
structs.h		
	Estruturas do programa	19

Chapter 3

Data Structure Documentation

3.1 `_listMachine` Struct Reference

Estrutura que representa a lista das máquinas.

```
#include <structs.h>
```

Data Fields

- int **time**
- int **id_mac**
- struct `_listMachine` * **next**
- struct `_listMachine` * **previous**

3.1.1 Detailed Description

Estrutura que representa a lista das máquinas.

The documentation for this struct was generated from the following file:

- [structs.h](#)

3.2 `_listMacOp` Struct Reference

Estrutura intermedia entre a lista das maquinas e das operações.

```
#include <structs.h>
```

Data Fields

- int **id_op**
- int **id_mac**
- struct `_listMacOp` * **next**
- struct `_listMacOp` * **previous**

3.2.1 Detailed Description

Estrutura intermedia entre a lista das maquinas e das operações.

The documentation for this struct was generated from the following file:

- [structs.h](#)

3.3 _listOperation Struct Reference

Estrutura que representa a lista das operações.

```
#include <structs.h>
```

Data Fields

- int **id_op**
- struct [_listOperation](#) * **next**
- struct [_listOperation](#) * **previous**

3.3.1 Detailed Description

Estrutura que representa a lista das operações.

The documentation for this struct was generated from the following file:

- [structs.h](#)

Chapter 4

File Documentation

4.1 functions.c File Reference

Funções do programa.

```
#include <stdio.h>
#include <stdlib.h>
#include "functions.h"
```

Functions

- [Machine](#) * [readMachines](#) ([Machine](#) *listMachine)
Leitura do ficheiro machines.txt e armazenamento em memória.
- [Machine](#) * [writeMachines](#) ([Machine](#) *listMachine)
Escrita do ficheiro machines.txt e armazenamento em memória.
- [MacOp](#) * [readMacOp](#) ([MacOp](#) *listMacOp)
Leitura do ficheiro macop.txt e armazenamento em memória.
- [MacOp](#) * [writeMacOp](#) ([MacOp](#) *listMacOp)
Escrita do ficheiro macop.txt e armazenamento em memória.
- [Operation](#) * [readOperations](#) ([Operation](#) *listOperation)
Leitura do ficheiro operations.txt e armazenamento em memória.
- [Operation](#) * [writeOperations](#) ([Operation](#) *listOperation)
Escrita do ficheiro operations.txt e armazenamento em memória.
- void [freeMachines](#) ([Machine](#) *list)
Desalocar memória ocupada pela lista das máquinas.
- void [freeMacOp](#) ([MacOp](#) *list)
Desalocar memória ocupada pela lista MacOp.
- void [freeOperations](#) ([Operation](#) *list)
Desalocar memória ocupada pela lista das operações.
- [Machine](#) * [head_insert_machine](#) ([Machine](#) *list, [Machine](#) *aux)
Inserção á cabeça na lista das máquinas.
- [MacOp](#) * [head_insert_macop](#) ([MacOp](#) *list, [MacOp](#) *aux)
Inserção á cabeça na lista MacOp.
- [Operation](#) * [head_insert_operation](#) ([Operation](#) *list, [Operation](#) *aux)

- Inserção á cabeça na lista das operações.*
- int `exist_mac` (`Machine` **listMachine, int id)
 - Verifica se um id de uma maquina inserido existe na lista das maquinas.*
- int `exist_op` (`Operation` **listOperation, int id)
 - Verifica se um id de uma operação inserido existe na lista das operações.*
- int `exist_MacOp` (`MacOp` **listMacOp, int id_mac, int id_op)
 - Verifica se uma combinação de id_mac e id_op existe na lista MacOp.*
- int `get_max_time` (`Machine` **listMachine)
 - Busca o valor maximo para a variavel tempo na lista das máquinas.*
- void `showMachineList` (`Machine` *list)
 - Listagem das máquinas na lista.*
- void `showMacOpList` (`MacOp` *list)
 - listagem da lista intermedia entre as operações e as máquinas*
- void `showOperationList` (`Operation` *list)
 - Listagem das operações na lista.*
- void `remove_MacOp` (`MacOp` **listMacOp, int id_op)
 - Remoção de uma operação da lista intermedia MacOp.*
- void `insert_operation` (`Operation` **listOperation, `MacOp` **listMacOp, `Machine` *listMachine)
 - Inserção de uma operação na lista das operações e lista MacOp.*
- void `remove_operation` (`Operation` **listOperation, `MacOp` **listMacOp, int id_op)
 - Remoção de uma operação da lista operações e respetivamente da lista intermedia MacOp.*
- void `change_operation` (`MacOp` **listMacOp, `Machine` *listMachine, int id_op)
 - Alteração de uma operação da lista operações.*
- float `avg_time` (`MacOp` **listMacOp, `Machine` **listMachine, int id_op)
 - Determinação da quantidade média de unidades de tempo necessárias para completar uma operação, considerando todas as alternativas possíveis.*
- int `min_time` (`Operation` **listOperation, `MacOp` **listMacOp, `Machine` **listMachine)
 - Determinação da quantidade mínima de unidades de tempo necessárias para completar o job e listagem das respetivas operações.*
- int `max_time` (`Operation` **listOperation, `MacOp` **listMacOp, `Machine` **listMachine)
 - Determinação da quantidade máxima de unidades de tempo necessárias para completar o job e listagem das respetivas operações.*
- void `interface_job` ()
 - Texto apresentado ao utilizador do menu job.*
- void `menu_job` (`Operation` **listOperation, `MacOp` **listMacOp, `Machine` **listMachine)
 - Menu de um determinado job selecionado.*

4.1.1 Detailed Description

Funções do programa.

Author

João Apresentação (a21152@alunos.ipca.pt)

Version

0.1

Date

2022-03-26

Copyright

Copyright (c) 2022

4.1.2 Function Documentation

4.1.2.1 avg_time()

```
float avg_time (
    MacOp ** listMacOp,
    Machine ** listMachine,
    int id_op )
```

Determinação da quantidade média de unidades de tempo necessárias para completar uma operação, considerando todas as alternativas possíveis.

Parameters

<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>listMachine</i>	lista das máquinas
<i>id_op</i>	id da operação

4.1.2.2 change_operation()

```
void change_operation (
    MacOp ** listMacOp,
    Machine * listMachine,
    int id_op )
```

Alteração de uma operação da lista operações.

Parameters

<i>listOperation</i>	lista das operações
<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>id_op</i>	id da operação a ser alterada

4.1.2.3 exist_mac()

```
int exist_mac (
    Machine ** listMachine,
    int id )
```

Verifica se um id de uma maquina inserido existe na lista das maquinas.

Parameters

<i>listMachine</i>	lista das máquinas
<i>id</i>	id de uma máquina inserido

Returns

int

4.1.2.4 exist_MacOp()

```
int exist_MacOp (
    MacOp ** listMacOp,
    int id_mac,
    int id_op )
```

Verifica se uma combinação de id_mac e id_op existe na lista MacOp.

Parameters

<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>id_mac</i>	id da maquina
<i>id_op</i>	id da operação

Returns

int

4.1.2.5 exist_op()

```
int exist_op (
    Operation ** listOperation,
    int id )
```

Verifica se um id de uma operação inserido existe na lista das operações.

Parameters

<i>listOperation</i>	lista das operações
<i>id</i>	id de uma operação inserido

Returns

int

4.1.2.6 freeMachines()

```
void freeMachines (
    Machine * list )
```

Desalocar memória ocupada pela lista das máquinas.

Parameters

<i>list</i>	lista das máquinas
-------------	--------------------

4.1.2.7 freeMacOp()

```
void freeMacOp (
    MacOp * list )
```

Desalocar memória ocupada pela lista MacOp.

Parameters

<i>list</i>	lista MacOp
-------------	-------------

4.1.2.8 freeOperations()

```
void freeOperations (
    Operation * list )
```

Desalocar memória ocupada pela lista das operações.

Parameters

<i>list</i>	lista das operações
-------------	---------------------

4.1.2.9 get_max_time()

```
int get_max_time (
    Machine ** listMachine )
```

Busca o valor maximo para a variavel tempo na lista das máquinas.

Parameters

<i>listMachine</i>	lista das máquinas
--------------------	--------------------

Returns

int

4.1.2.10 head_insert_machine()

```
Machine * head_insert_machine (
    Machine * list,
    Machine * aux )
```

Inserção á cabeça na lista das máquinas.

Parameters

<i>list</i>	Lista das máquinas
<i>aux</i>	Lista auxiliar

Returns

Machine*

4.1.2.11 head_insert_macop()

```
MacOp * head_insert_macop (
    MacOp * list,
    MacOp * aux )
```

Inserção á cabeça na lista MacOp.

Parameters

<i>list</i>	Lista MacOp
<i>aux</i>	Lista auxiliar

Returns

MacOp*

4.1.2.12 head_insert_operation()

```
Operation * head_insert_operation (
    Operation * list,
    Operation * aux )
```

Inserção á cabeça na lista das operações.

Parameters

<i>list</i>	Lista de operações
<i>aux</i>	Lista auxiliar

Returns

Operation*

4.1.2.13 insert_operation()

```
void insert_operation (
    Operation ** listOperation,
    MacOp ** listMacOp,
    Machine * listMachine )
```

Inserção de uma operação na lista das operações e lista MacOp.

Parameters

<i>listOperation</i>	lista das operações
<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>listMachine</i>	lista das máquinas

4.1.2.14 max_time()

```
int max_time (
    Operation ** listOperation,
    MacOp ** listMacOp,
    Machine ** listMachine )
```

Determinação da quantidade máxima de unidades de tempo necessárias para completar o job e listagem das respetivas operações.

Parameters

<i>listOperation</i>	lista das operações
<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>listMachine</i>	lista das máquinas

4.1.2.15 menu_job()

```
void menu_job (
    Operation ** listOperation,
    MacOp ** listMacOp,
    Machine ** listMachine )
```

Menu de um determinado job selecionado.

Parameters

<i>listOperation</i>	lista das operações
<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>listMachine</i>	lista das máquinas

4.1.2.16 min_time()

```
int min_time (
    Operation ** listOperation,
    MacOp ** listMacOp,
    Machine ** listMachine )
```

Determinação da quantidade mínima de unidades de tempo necessárias para completar o job e listagem das respetivas operações.

Parameters

<i>listOperation</i>	lista das operações
<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>listMachine</i>	lista das máquinas

4.1.2.17 readMachines()

```
Machine * readMachines (
    Machine * listMachine )
```

Leitura do ficheiro machines.txt e armazenamento em memória.

Parameters

<i>listMachine</i>	Lista que contem todas as máquinas
--------------------	------------------------------------

Returns

Machine*

4.1.2.18 readMacOp()

```
MacOp * readMacOp (
    MacOp * listMacOp )
```

Leitura do ficheiro macop.txt e armazenamento em memória.

Parameters

<i>listMacOp</i>	Lista que contem os ids de todas as operações e suas respectivas máquinas
------------------	---

Returns

MacOp*

4.1.2.19 readOperations()

```
Operation * readOperations (
    Operation * listOperation )
```

Leitura do ficheiro operations.txt e armazenamento em memória.

Parameters

<i>listOperation</i>	Lista que contem todas as operações
----------------------	-------------------------------------

Returns

Operation*

4.1.2.20 remove_MacOp()

```
void remove_MacOp (
    MacOp ** listMacOp,
    int id_op )
```

Remoção de uma operação da lista intermedia MacOp.

Parameters

<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>id_op</i>	id da operação a ser removida

4.1.2.21 remove_operation()

```
void remove_operation (
    Operation ** listOperation,
    MacOp ** listMacOp,
    int id_op )
```

Remoção de uma operação da lista operações e respetivamente da lista intermedia MacOp.

Parameters

<i>listOperation</i>	lista das operações
<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
<i>id_op</i>	id da operação a ser removida

4.1.2.22 showMachineList()

```
void showMachineList (
    Machine * list )
```

Listagem das máquinas na lista.

Parameters

<i>list</i>	Lista das máquinas
-------------	--------------------

4.1.2.23 showMacOpList()

```
void showMacOpList (
    MacOp * list )
```

listagem da lista intermedia entre as operações e as máquinas

Parameters

<i>list</i>	Lista MacOp
-------------	-------------

4.1.2.24 showOperationList()

```
void showOperationList (
    Operation * list )
```

Listagem das operações na lista.

Parameters

<i>list</i>	Lista das operações
-------------	---------------------

4.1.2.25 writeMachines()

```
Machine * writeMachines (
    Machine * listMachine )
```

Escrita do ficheiro machines.txt e armazenamento em memória.

Parameters

<i>listMachine</i>	lista das maquinas
--------------------	--------------------

Returns

Machine*

4.1.2.26 writeMacOp()

```
MacOp * writeMacOp (
    MacOp * listMacOp )
```

Escrita do ficheiro macop.txt e armazenamento em memória.

Parameters

<i>listMacOp</i>	lista intermedia entre as operações e as máquinas
------------------	---

Returns

MacOp*

4.1.2.27 writeOperations()

```
Operation * writeOperations (
    Operation * listOperation )
```

Escrita do ficheiro operations.txt e armazenamento em memória.

Parameters

<i>listOperation</i>	lista das operações
----------------------	---------------------

Returns

Operation*

4.2 functions.h

```
1
12 #include "structs.h"
13
14 Machine *readMachines(Machine *listMachine);
15 Machine *writeMachines(Machine *listMachine);
16 void freeMachines(Machine *list);
17 Machine *head_insert_machine(Machine *list, Machine *aux);
18 void showMachineList(Machine *list);
19
20 MacOp *readMacOp(MacOp *listMacOp);
21 MacOp *writeMacOp(MacOp *listMacOp);
22 void freeMacOp(MacOp *list);
23 MacOp *head_insert_macop(MacOp *list, MacOp *aux);
24 void showMacOpList(MacOp *list);
25
26 Operation *readOperations(Operation *listOperation);
27 Operation *writeOperations(Operation *listOperation);
28 void freeOperations(Operation *list);
29 Operation *head_insert_operation(Operation *list, Operation *aux);
30 void showOperationList(Operation *list);
31
32 void interface_job();
33 void menu_job(Operation **listOperation, MacOp **listMacOp, Machine **listMachine);
34
35 void remove_MacOp(MacOp **listMacOp, int id_op);
36
37 void insert_operation(Operation **listOperation, MacOp **listMacOp, Machine *listMachine);
38 void remove_operation(Operation **listOperation, MacOp **listMacOp, int id_op);
39 void change_operation(MacOp **listMacOp, Machine *listMachine, int id_op);
40
41 int min_time(Operation **listOperation, MacOp **listMacOp, Machine **listMachine);
42 int max_time(Operation **listOperation, MacOp **listMacOp, Machine **listMachine);
43 float avg_time(MacOp **listMacOp, Machine **listMachine, int id_op);
44
45 int exist_mac(Machine **listMachine, int id);
46 int exist_op(Operation **listOperation, int id);
47 int exist_MacOp(MacOp **listMacOp, int id_mac, int id_op);
48 int get_max_time(Machine **listMachine);
```

4.3 main.c File Reference

Solução para problema de escalonamento (Flexifile job shop problem)

```
#include <stdio.h>
#include <stdlib.h>
#include "functions.h"
```

Functions

- `int main ()`
Função main.

4.3.1 Detailed Description

Solução para problema de escalonamento (Flexifile job shop problem)

Author

João Apresentação (a21152@alunos.ipca.pt)

Version

0.1

Date

2022-03-26

Copyright

Copyright (c) 2022

4.3.2 Function Documentation

4.3.2.1 main()

```
int main ( )
```

Função main.

Returns

int

4.4 structs.h File Reference

Estruturas do programa.

Data Structures

- struct [_listMachine](#)
Estrutura que representa a lista das máquinas.
- struct [_listMacOp](#)
Estrutura intermedia entre a lista das maquinas e das operações.
- struct [_listOperation](#)
Estrutura que representa a lista das operações.

Typedefs

- typedef struct [_listMachine](#) **Machine**
Estrutura que representa a lista das máquinas.
- typedef struct [_listMacOp](#) **MacOp**
Estrutura intermedia entre a lista das maquinas e das operações.
- typedef struct [_listOperation](#) **Operation**
Estrutura que representa a lista das operações.

4.4.1 Detailed Description

Estruturas do programa.

Author

João Apresentação (a21152@alunos.ipca.pt)

Version

0.1

Date

2022-03-26

Copyright

Copyright (c) 2022

4.5 structs.h

[Go to the documentation of this file.](#)

```
1
16 typedef struct _listMachine
17 {
18     int time, id_mac;
19     struct _listMachine *next, *previous;
20 } Machine;
21
26 typedef struct _listMacOp
27 {
28     int id_op, id_mac;
29     struct _listMacOp *next, *previous;
30 } MacOp;
31
36 typedef struct _listOperation
37 {
38     int id_op;
39     struct _listOperation *next, *previous;
40 } Operation;
```


Index

- [_listMacOp, 5](#)
 - [_listMachine, 5](#)
 - [_listOperation, 6](#)
- [avg_time](#)
 - [functions.c, 9](#)
- [change_operation](#)
 - [functions.c, 9](#)
- [exist_mac](#)
 - [functions.c, 9](#)
- [exist_MacOp](#)
 - [functions.c, 10](#)
- [exist_op](#)
 - [functions.c, 10](#)
- [freeMachines](#)
 - [functions.c, 11](#)
- [freeMacOp](#)
 - [functions.c, 11](#)
- [freeOperations](#)
 - [functions.c, 11](#)
- [functions.c, 7](#)
 - [avg_time, 9](#)
 - [change_operation, 9](#)
 - [exist_mac, 9](#)
 - [exist_MacOp, 10](#)
 - [exist_op, 10](#)
 - [freeMachines, 11](#)
 - [freeMacOp, 11](#)
 - [freeOperations, 11](#)
 - [get_max_time, 11](#)
 - [head_insert_machine, 12](#)
 - [head_insert_macop, 12](#)
 - [head_insert_operation, 12](#)
 - [insert_operation, 13](#)
 - [max_time, 13](#)
 - [menu_job, 14](#)
 - [min_time, 14](#)
 - [readMachines, 14](#)
 - [readMacOp, 15](#)
 - [readOperations, 15](#)
 - [remove_MacOp, 15](#)
 - [remove_operation, 16](#)
 - [showMachineList, 16](#)
 - [showMacOpList, 16](#)
 - [showOperationList, 17](#)
 - [writeMachines, 17](#)
 - [writeMacOp, 17](#)
 - [writeOperations, 17](#)
- [get_max_time](#)
 - [functions.c, 11](#)
- [head_insert_machine](#)
 - [functions.c, 12](#)
- [head_insert_macop](#)
 - [functions.c, 12](#)
- [head_insert_operation](#)
 - [functions.c, 12](#)
- [insert_operation](#)
 - [functions.c, 13](#)
- [main](#)
 - [main.c, 19](#)
- [main.c, 18](#)
 - [main, 19](#)
- [max_time](#)
 - [functions.c, 13](#)
- [menu_job](#)
 - [functions.c, 14](#)
- [min_time](#)
 - [functions.c, 14](#)
- [readMachines](#)
 - [functions.c, 14](#)
- [readMacOp](#)
 - [functions.c, 15](#)
- [readOperations](#)
 - [functions.c, 15](#)
- [remove_MacOp](#)
 - [functions.c, 15](#)
- [remove_operation](#)
 - [functions.c, 16](#)
- [showMachineList](#)
 - [functions.c, 16](#)
- [showMacOpList](#)
 - [functions.c, 16](#)
- [showOperationList](#)
 - [functions.c, 17](#)
- [structs.h, 19](#)
- [writeMachines](#)
 - [functions.c, 17](#)
- [writeMacOp](#)
 - [functions.c, 17](#)
- [writeOperations](#)
 - [functions.c, 17](#)