

## MINI SHELL - PROJET

Generated by Doxygen 1.9.1



<b>1 Class Index</b>	<b>1</b>
1.1 Class List	1
<b>2 File Index</b>	<b>3</b>
2.1 File List	3
<b>3 Class Documentation</b>	<b>5</b>
3.1 cmdline Struct Reference	5
3.2 FinJobs Struct Reference	5
3.3 Jobs Struct Reference	6
3.4 rio_t Struct Reference	6
<b>4 File Documentation</b>	<b>7</b>
4.1 includes/externCmd.h File Reference	7
4.1.1 Detailed Description	7
4.1.2 Function Documentation	7
4.1.2.1 commandeExterne()	7
4.2 includes/internCmd.h File Reference	8
4.2.1 Detailed Description	9
4.2.2 Function Documentation	9
4.2.2.1 commandeInterne()	9
4.2.2.2 isChangeDir()	9
4.2.2.3 isJobs()	11
4.2.2.4 isQuitShell()	11
4.2.2.5 printShellEnv()	12
4.3 includes/jobs.h File Reference	12
4.3.1 Detailed Description	13
4.3.2 Typedef Documentation	13
4.3.2.1 FinJobs	13
4.3.2.2 Jobs	14
4.3.3 Function Documentation	14
4.3.3.1 addFinJob()	14
4.3.3.2 addJob()	14
4.3.3.3 deleteFinJob()	15
4.3.3.4 deleteJob()	15
4.3.3.5 findJobByName()	15
4.3.3.6 findJobByPID()	16
4.3.3.7 printFinJobs()	16
4.3.3.8 printJobs()	16
4.4 includes/redirection.h File Reference	17
4.4.1 Detailed Description	17
4.4.2 Function Documentation	17
4.4.2.1 redirectionEntree()	17

4.4.2.2 redirectionSortie() . . . . .	18
4.5 includes/signalHandlers.h File Reference . . . . .	18
4.5.1 Detailed Description . . . . .	18
<b>Index</b>	<b>19</b>

# Chapter 1

## Class Index

### 1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">cmdline</a>	.....	5
<a href="#">FinJobs</a>	.....	5
<a href="#">Jobs</a>	.....	6
<a href="#">rio_t</a>	.....	6



## Chapter 2

# File Index

### 2.1 File List

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

includes/ <b>csapp.h</b>	??
includes/ <a href="#">externCmd.h</a>	7
includes/ <a href="#">internCmd.h</a>	8
includes/ <a href="#">jobs.h</a>	12
includes/ <b>readcmd.h</b>	??
includes/ <a href="#">redirection.h</a>	17
includes/ <a href="#">signalHandlers.h</a>	18





## Chapter 3

# Class Documentation

### 3.1 cmdline Struct Reference

#### Public Attributes

- char \* **err**
- char \* **in**
- char \* **out**
- char \*\*\* **seq**
- int **bg**

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

- `includes/readcmd.h`

### 3.2 FinJobs Struct Reference

Collaboration diagram for FinJobs:



#### Public Attributes

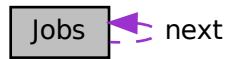
- int **pid**
- [FinJobs](#) \* **next**

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

- `includes/jobs.h`

### 3.3 Jobs Struct Reference

Collaboration diagram for Jobs:



#### Public Attributes

- char \* **name**
- char \* **state**
- int **pid**
- int **num**
- [Jobs](#) \* **next**

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

- [includes/jobs.h](#)

### 3.4 rio\_t Struct Reference

#### Public Attributes

- int **rio\_fd**
- int **rio\_cnt**
- char \* **rio\_bufptr**
- char **rio\_buf** [RIO\_BUFSIZE]

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

- [includes/csapp.h](#)

# Chapter 4

## File Documentation

### 4.1 includes/externCmd.h File Reference

#### Functions

- int [commandeExterne](#) (char \*\*cmd, char \*inNom, char \*outNom, int pipes[2][2], int deb, int fin)  
*Gestionnaire des commandes externes.*

#### 4.1.1 Detailed Description

##### Author

@OliverMineau  
@alexisthach

##### Version

0.1

##### Date

2023-03-01

##### Copyright

Copyright (c) 2023

#### 4.1.2 Function Documentation

##### 4.1.2.1 commandeExterne()

```
int commandeExterne (
    char ** cmd,
    char * inNom,
    char * outNom,
    int pipes[2][2],
    int deb,
    int fin )
```

Gestionnaire des commandes externes.

## Parameters

<i>cmd</i>	Contient une commande et ses parametres.
<i>inNom</i>	Nom du fichier donné en entrée si fournit, sinon null.
<i>outNom</i>	Nom du fichier donné en sortie si fournit, sinon null.
<i>pipes</i>	Tableau de deux pipes contenant prevCmdPipe et nextCmdPipe.
<i>deb</i>	Indique si la commande lue est la premiere.
<i>fin</i>	Indique si la commande lue est la derniere.

## Returns

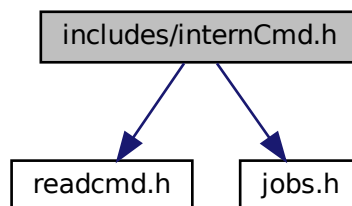
int 1 si erreur, 0 sinon.

## 4.2 includes/internCmd.h File Reference

```
#include "readcmd.h"
```

```
#include "jobs.h"
```

Include dependency graph for internCmd.h:



## Functions

- void [printShellEnv](#) ()  
*Affiche l'environnement courant du shell.*
- void [isQuitShell](#) (struct [cmdline](#) \*cmd, int n)  
*Vérifie si la commande donnée doit fermer le shell.*
- int [isChangeDir](#) (struct [cmdline](#) \*cmd, int n)  
*Vérifie si la commande donnée doit changer l'environnement du shell.*
- int [isJobs](#) (struct [cmdline](#) \*cmd, int n, [Jobs](#) \*jobs)  
*Vérifie si la commande donnée doit afficher les jobs en cours.*
- int [commandeInterne](#) (struct [cmdline](#) \*cmd, int n, [Jobs](#) \*jobs)  
*Gestionnaire des commandes internes.*

## 4.2.1 Detailed Description

### Author

@OliverMineau

@alexisthach

### Version

0.1

### Date

2023-03-01

### Copyright

Copyright (c) 2023

## 4.2.2 Function Documentation

### 4.2.2.1 commandeInterne()

```
int commandeInterne (
    struct cmdline * cmd,
    int n,
    Jobs * jobs )
```

Gestionnaire des commandes internes.

#### Parameters

<i>cmd</i>	Structure cmdline contenant les commandes.
<i>n</i>	Position/Numéro de la commande à vérifier.
<i>jobs</i>	Liste des jobs en cours.

#### Returns

int

### 4.2.2.2 isChangeDir()

```
int isChangeDir (
    struct cmdline * cmd,
    int n )
```

Vérifie si la commande donnée doit changer l'environnement du shell.

**Parameters**

<i>cmd</i>	Structure cmdline contenant les commandes.
<i>n</i>	Position/Numéro de la commande à vérifier.

**Returns**

int 1 si commande ne peut etre suivie par d'autres commandes, 0 sinon.

**4.2.2.3 isJobs()**

```
int isJobs (
    struct cmdline * cmd,
    int n,
    Jobs * jobs )
```

Vérifie si la commande donnée doit afficher les jobs en cours.

**Parameters**

<i>cmd</i>	Structure cmdline contenant les commandes.
<i>n</i>	Position/Numéro de la commande à vérifier.
<i>jobs</i>	Liste des jobs en cours.

**Returns**

int 1 si commande ne peut etre suivie par d'autres commandes, 0 sinon.

**4.2.2.4 isQuitShell()**

```
void isQuitShell (
    struct cmdline * cmd,
    int n )
```

Vérifie si la commande donnée doit fermer le shell.

**Parameters**

<i>cmd</i>	Structure cmdline contenant les commandes.
<i>n</i>	Position/Numéro de la commande à vérifier.

#### 4.2.2.5 printShellEnv()

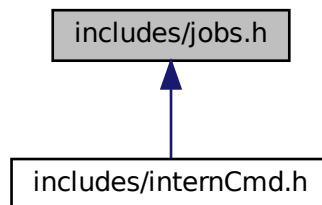
```
void printShellEnv ( )
```

Affiche l'environnement courant du shell.

Trouver le Current Working Directory `strlen(getenv("HOME"))` : on trouve la taille de l'env et affiche apres

### 4.3 includes/jobs.h File Reference

This graph shows which files directly or indirectly include this file:



#### Classes

- struct [Jobs](#)
- struct [FinJobs](#)

#### Typedefs

- typedef struct [Jobs](#) [Jobs](#)  
*Structure comportant les informations sur un job.*
- typedef struct [FinJobs](#) [FinJobs](#)  
*Structure comportant les informations sur un job terminé.*

#### Functions

- void [addJob](#) ([Jobs](#) \*\*jobs, char \*\*name, int pid, char \*state)  
*Ajout d'un job dans la liste.*
- [Jobs](#) \* [findJobByName](#) ([Jobs](#) \*jobs, char \*name)  
*Trouve si le job de nom name existe dans la liste.*
- [Jobs](#) \* [findJobByPID](#) ([Jobs](#) \*jobs, int pid)  
*Trouve si le job de pid pid existe dans la liste.*
- int [deleteJob](#) ([Jobs](#) \*\*jobs, [Jobs](#) \*del)  
*Supprimer un job par pid.*



- void `printJobs` (`Jobs *jobs`)  
*Affiche les jobs en executions.*
- void `addFinJob` (`FinJobs **fj`, int pid)  
*Add le pid d'un job terminé dans la liste.*
- void `deleteFinJob` (`FinJobs **fj`, int pid)  
*Supprime le pid d'un job terminé dans la liste.*
- void `printFinJobs` (`FinJobs *jobs`)  
*Affiche les jobs terminés et non tués.*

### 4.3.1 Detailed Description

#### Author

@OliverMineau  
@alexisthach

#### Version

0.1

#### Date

2023-03-01

#### Copyright

Copyright (c) 2023

### 4.3.2 Typedef Documentation

#### 4.3.2.1 FinJobs

```
typedef struct FinJobs FinJobs
```

Structure comportant les informations sur un job terminé.

#### Parameters

<i>pid</i>	Pid du processus terminé à tuer.
<i>next</i>	Job suivant dans la liste.

#### 4.3.2.2 Jobs

```
typedef struct Jobs Jobs
```

Structure comportant les informations sur un job.

##### Parameters

<i>name</i>	Nom de la commande executée.
<i>state</i>	Etat du job : Running, Stopped.
<i>pid</i>	Pid du processus executant le job.
<i>num</i>	Numéro du job.
<i>next</i>	Job suivant dans la liste.

### 4.3.3 Function Documentation

#### 4.3.3.1 addFinJob()

```
void addFinJob (  
    FinJobs ** fj,  
    int pid )
```

Add le pid d'un job terminé dans la liste.

##### Parameters

<i>fj</i>	Liste des jobs terminés.
<i>pid</i>	Pid du job à ajouter.

#### 4.3.3.2 addJob()

```
void addJob (  
    Jobs ** jobs,  
    char ** name,  
    int pid,  
    char * state )
```

Ajout d'un job dans la liste.

##### Parameters

<i>jobs</i>	Job à ajouter.
<i>name</i>	Nom de la commande executée.
<i>pid</i>	Pid de la commande executée.
<i>state</i>	Etat de la commande executée.

#### 4.3.3.3 deleteFinJob()

```
void deleteFinJob (
    FinJobs ** fj,
    int pid )
```

Supprime le pid d'un job terminé dans la liste.

##### Parameters

<i>fj</i>	Liste des jobs terminés.
<i>pid</i>	Pid à supprimer.

#### 4.3.3.4 deleteJob()

```
int deleteJob (
    Jobs ** jobs,
    Jobs * del )
```

Supprimer un job par pid.

##### Parameters

<i>jobs</i>	Liste des jobs.
<i>del</i>	Job a supprimer.

##### Returns

int 0 si job pas supprimé, 1 sinon.

#### 4.3.3.5 findJobByName()

```
Jobs* findJobByName (
    Jobs * jobs,
    char * name )
```

Trouve si le job de nom name existe dans la liste.

##### Parameters

<i>jobs</i>	Liste des jobs.
<i>name</i>	Nom du job à chercher.

**Returns**

Jobs\* Job trouvé.

**4.3.3.6 findJobByPID()**

```
Jobs* findJobByPID (
    Jobs * jobs,
    int pid )
```

Trouve si le job de pid pid existe dans la liste.

**Parameters**

<i>jobs</i>	Liste des jobs.
<i>pid</i>	Pid du job à chercher.

**Returns**

Jobs\* Job trouvé.

**4.3.3.7 printFinJobs()**

```
void printFinJobs (
    FinJobs * jobs )
```

Affiche les jobs terminés et non tués.

**Parameters**

<i>jobs</i>	Liste des jobs.
-------------	-----------------

**4.3.3.8 printJobs()**

```
void printJobs (
    Jobs * jobs )
```

Affiche les jobs en executions.

**Parameters**

<i>jobs</i>	Liste des jobs.
-------------	-----------------

## 4.4 includes/redirection.h File Reference

### Functions

- int [redirectionEntree](#) (char \*inNom, int \*prevCmdPipe, int deb, int \*fd\_in)  
*Redirige l'entrée si un fichier d'entrée est fournit.*
- int [redirectionSortie](#) (char \*outNom, int \*nextCmdPipe, int fin, int deb, int \*fd\_out)  
*Redirige la sortie si un fichier de sortie est fournit.*

### 4.4.1 Detailed Description

#### Author

@OliverMineau  
@alexisthach

#### Version

0.1

#### Date

2023-03-01

#### Copyright

Copyright (c) 2023

### 4.4.2 Function Documentation

#### 4.4.2.1 redirectionEntree()

```
int redirectionEntree (  
    char * inNom,  
    int * prevCmdPipe,  
    int deb,  
    int * fd_in )
```

Redirige l'entrée si un fichier d'entrée est fournit.

#### Parameters

<i>inNom</i>	Nom du fichier donnée en entrée.
<i>prevCmdPipe</i>	Pipe venant de la commande précédente.
<i>deb</i>	Indique si la commande est la première d'une sequence de commandes.
<i>fd_in</i>	Descripteur de fichier correspondant au fichier de nom inNom.

**Returns**

int 1 si erreur, 0 sinon.

**4.4.2.2 redirectionSortie()**

```
int redirectionSortie (
    char * outNom,
    int * nextCmdPipe,
    int fin,
    int deb,
    int * fd_out )
```

Redirige la sortie si un fichier de sortie est fourni.

**Parameters**

<i>outNom</i>	Nom du fichier donnée en sortie.
<i>nextCmdPipe</i>	Pipe allant vers la prochaine commande.
<i>fin</i>	Indique si la commande est la dernière d'une sequence de commandes.
<i>deb</i>	Indique si la commande est la première d'une sequence de commandes.
<i>fd_out</i>	Descripteur de fichier correspondant au fichier de nom outNom.

**Returns**

int 1 si erreur, 0 sinon.

**4.5 includes/signalHandlers.h File Reference****Functions**

- void **ctrlCHandler** (int sig)
- void **ctrlZHandler** (int sig)

**4.5.1 Detailed Description****Author**

@OliverMineau  
@alexisthach

**Version**

0.1

**Date**

2023-03-01

**Copyright**

Copyright (c) 2023

# Index

- addFinJob
  - jobs.h, [14](#)
- addJob
  - jobs.h, [14](#)
- cmdline, [5](#)
- commandeExterne
  - externCmd.h, [7](#)
- commandeInterne
  - internCmd.h, [9](#)
- deleteFinJob
  - jobs.h, [15](#)
- deleteJob
  - jobs.h, [15](#)
- externCmd.h
  - commandeExterne, [7](#)
- findJobByName
  - jobs.h, [15](#)
- findJobByPID
  - jobs.h, [16](#)
- FinJobs, [5](#)
  - jobs.h, [13](#)
- includes/externCmd.h, [7](#)
- includes/internCmd.h, [8](#)
- includes/jobs.h, [12](#)
- includes/redirection.h, [17](#)
- includes/signalHandlers.h, [18](#)
- internCmd.h
  - commandeInterne, [9](#)
  - isChangeDir, [9](#)
  - isJobs, [11](#)
  - isQuitShell, [11](#)
  - printShellEnv, [11](#)
- isChangeDir
  - internCmd.h, [9](#)
- isJobs
  - internCmd.h, [11](#)
- isQuitShell
  - internCmd.h, [11](#)
- Jobs, [6](#)
  - jobs.h, [13](#)
- jobs.h
  - addFinJob, [14](#)
  - addJob, [14](#)
  - deleteFinJob, [15](#)
  - deleteJob, [15](#)
  - findJobByName, [15](#)
  - findJobByPID, [16](#)
  - FinJobs, [13](#)
  - Jobs, [13](#)
  - printFinJobs, [16](#)
  - printJobs, [16](#)
- printFinJobs
  - jobs.h, [16](#)
- printJobs
  - jobs.h, [16](#)
- printShellEnv
  - internCmd.h, [11](#)
- redirection.h
  - redirectionEntree, [17](#)
  - redirectionSortie, [18](#)
- redirectionEntree
  - redirection.h, [17](#)
- redirectionSortie
  - redirection.h, [18](#)
- rio\_t, [6](#)