

MINI SHELL - PROJET

Generated by Doxygen 1.9.1

1 Mini projet SHELL	1
1.1 Compilation	1
1.2 Utilisation du programme	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 cmdline Struct Reference	7
4.2 Jobs Struct Reference	7
4.3 rio_t Struct Reference	8
5 File Documentation	9
5.1 includes/externCmd.h File Reference	9
5.1.1 Detailed Description	10
5.1.2 Function Documentation	10
5.1.2.1 commandeExterne()	10
5.1.2.2 seqToChar()	11
5.2 includes/internCmd.h File Reference	11
5.2.1 Detailed Description	12
5.2.2 Function Documentation	13
5.2.2.1 commandeInterne()	13
5.2.2.2 isBg()	13
5.2.2.3 isChangeDir()	14
5.2.2.4 isFg()	14
5.2.2.5 isJobs()	14
5.2.2.6 isQuitShell()	15
5.2.2.7 printShellEnv()	15
5.3 includes/jobs.h File Reference	16
5.3.1 Detailed Description	17
5.3.2 Typedef Documentation	17
5.3.2.1 Jobs	17
5.3.3 Function Documentation	18
5.3.3.1 addJob()	18
5.3.3.2 deleteJob()	18
5.3.3.3 findJobByName()	18
5.3.3.4 findJobByPID()	19
5.3.3.5 findJobInFG()	19
5.3.3.6 findJobNameNum()	20
5.3.3.7 jobEnFG()	20
5.3.3.8 printJobs()	20

5.4 includes/redirection.h File Reference	21
5.4.1 Detailed Description	21
5.4.2 Function Documentation	21
5.4.2.1 redirectionEntree()	21
5.4.2.2 redirectionSortie()	22
5.5 includes/shell.h File Reference	22
5.5.1 Detailed Description	23
5.6 includes/signalHandlers.h File Reference	23
5.6.1 Detailed Description	24
5.6.2 Function Documentation	24
5.6.2.1 childHandler()	24
5.6.2.2 ctrlCHandler()	25
5.6.2.3 ctrlZHandler()	25
Index	27

Chapter 1

Mini projet SHELL

SR - Mini projet SHELL, année 2022-2023 Réalisation d'un shell.

Utilisation

1.1 Compilation

Compilation simple : make

make <regle> <variable_env>=<valeur> Regles : "test" : Variable d'environnement : "T": Execute le test de nom test<valeur>.txt. "" : Variable d'environnement : "DEBUG": Affiche des infos utiles lors de l'execution du programme si <valeur>=1.

1.2 Utilisation du programme

./shell

Chapter 2

Class Index

2.1 Class List

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

cmdline	7
Jobs	7
rio_t	8

Chapter 3

File Index

3.1 File List

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

includes/ csapp.h	??
includes/ externCmd.h	9
includes/ internCmd.h	11
includes/ jobs.h	16
includes/ readcmd.h	??
includes/ redirection.h	21
includes/ shell.h	22
includes/ signalHandlers.h	23

Chapter 4

Class Documentation

4.1 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

4.2 Jobs Struct Reference

Collaboration diagram for Jobs:



Public Attributes

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

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

- [includes/jobs.h](#)

4.3 `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](#)

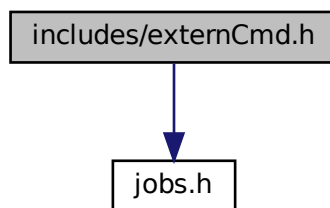
Chapter 5

File Documentation

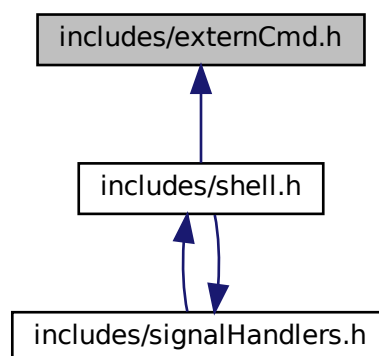
5.1 includes/externCmd.h File Reference

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

Include dependency graph for externCmd.h:



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



Functions

- int `commandeExterne` (struct `cmdline` *l, int n, int pipes[2][2], int deb, int fin, Jobs **jobs)
Gestionnaire des commandes externes.
- void `seqToChar` (struct `cmdline` *l, char *nomCommande)
Concatene toutes les commandes en une seule.

5.1.1 Detailed Description

Author

@OliverMineau

@alexisthach

Version

0.1

Date

2023-03-01

Copyright

Copyright (c) 2023

5.1.2 Function Documentation

5.1.2.1 `commandeExterne()`

```
int commandeExterne (
    struct cmdline * l,
    int n,
    int pipes[2][2],
    int deb,
    int fin,
    Jobs ** jobs )
```

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.

5.1.2.2 seqToChar()

```
void seqToChar (
    struct cmdline * l,
    char * nomCommande )
```

Concatene toutes les commandes en une seule.

Parameters

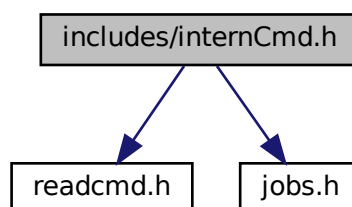
<i>l</i>	Contient toutes les commandes.
<i>nomCommande</i>	Contient les commandes concatenees.

5.2 includes/internCmd.h File Reference

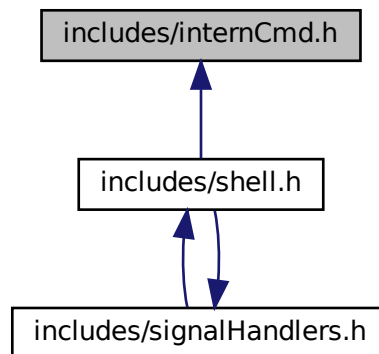
```
#include "readcmd.h"
```

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

Include dependency graph for internCmd.h:



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



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, char *outNom, int pipes[2][2], int deb, int fin)
Gestionnaire des commandes internes.
- int `isFg` (struct `cmdline` *cmd, int n, Jobs *jobs)
Vérifie si la commande donnée doit mettre en premier plan un job.
- int `isBg` (struct `cmdline` *cmd, int n, Jobs *jobs)
Vérifie si la commande donnée doit mettre en arriere plan un job.

5.2.1 Detailed Description

Author

@OliverMineau
@alexisthach

Version

0.1

Date

2023-03-01

Copyright

Copyright (c) 2023

5.2.2 Function Documentation

5.2.2.1 commandeInterne()

```
int commandeInterne (
    struct cmdline * cmd,
    int n,
    Jobs ** jobs,
    char * outNom,
    int pipes[2][2],
    int deb,
    int fin )
```

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 1 si c'est une commande interne, -1 si c'est une commande interne qui ne peut pas etre suivie d'une autre commande, 0 sinon.

5.2.2.2 isBg()

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

Vérifie si la commande donnée doit mettre en arriere plan un job.

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

5.2.2.3 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.

5.2.2.4 isFg()

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

Vérifie si la commande donnée doit mettre en premier plan un job.

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

5.2.2.5 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.

5.2.2.6 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.

5.2.2.7 printShellEnv()

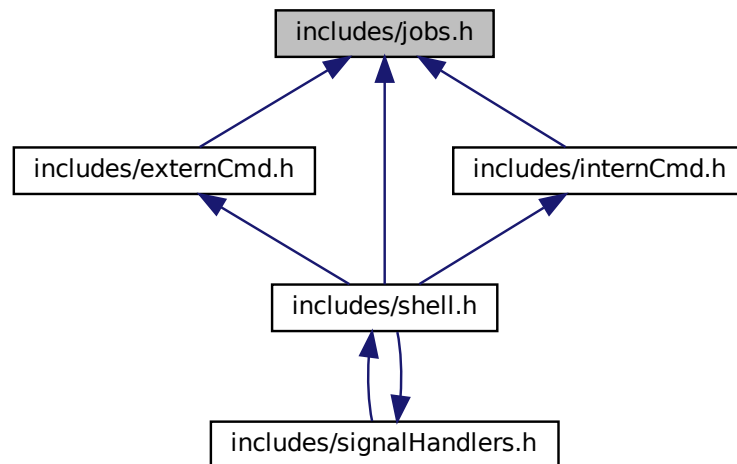
```
void printShellEnv ( )
```

Affiche l'environnement courant du shell.

Trouver le Current Working Directory getcwd donne le chemin complet, on enleve le chemin de home avec strlen(getenv("HOME")) : on trouve la taille de l'env et affiche a partir de là

5.3 includes/jobs.h File Reference

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



Classes

- struct [Jobs](#)

Typedefs

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

Enumerations

- enum [state](#) { **RUNNING** , **STOPPED** }
Etat d'un job.
- enum [ground](#) { **BACKGROUND** , **FOREGROUND** }
Plan d'un job.

Functions

- void [addJob](#) ([Jobs](#) **[jobs](#), char *name, int pid, enum [state](#) state, enum [ground](#) ground)
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.
- int `jobEnFG` (`Jobs *jobs`)
Indique si le job est en premier plan.
- `Jobs * findJobInFG` (`Jobs *jobs`)
Recherche un job en premier plan.
- `Jobs * findJobNameNum` (`Jobs *jobs`, `char *id`)
Trouve un job par son nom ou numero.

5.3.1 Detailed Description

Author

@OliverMineau

@alexisthach

Version

0.1

Date

2023-03-01

Copyright

Copyright (c) 2023

5.3.2 Typedef Documentation

5.3.2.1 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.
<i>bg</i>	1 si la commande est en background, 0 sinon.

5.3.3 Function Documentation

5.3.3.1 addJob()

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

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.

5.3.3.2 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.

5.3.3.3 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.

5.3.3.4 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.

5.3.3.5 findJobInFG()

```
Jobs* findJobInFG (
    Jobs * jobs )
```

Recherche un job en premier plan.

Parameters

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

Returns

Jobs*

5.3.3.6 findJobNameNum()

```
Jobs* findJobNameNum (
    Jobs * jobs,
    char * id )
```

Trouve un job par son nom ou numero.

Parameters

<i>jobs</i>	Liste des jobs.
<i>id</i>	Numero ou nom du job.

Returns

Jobs*

5.3.3.7 jobEnFG()

```
int jobEnFG (
    Jobs * jobs )
```

Indique si le job est en premier plan.

Parameters

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

Returns

int

5.3.3.8 printJobs()

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

Affiche les jobs en executions.

Parameters

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

5.4 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 fourni.
- int [redirectionSortie](#) (char *outNom, int *nextCmdPipe, int fin, int deb, int *fd_out)
Redirige la sortie si un fichier de sortie est fourni.

5.4.1 Detailed Description

Author

@OliverMineau
@alexisthach

Version

0.1

Date

2023-03-01

Copyright

Copyright (c) 2023

5.4.2 Function Documentation

5.4.2.1 redirectionEntree()

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

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

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.

5.4.2.2 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.

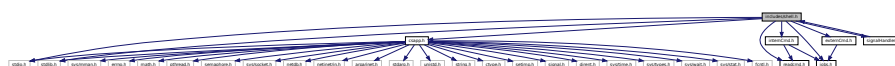
Returns

int 1 si erreur, 0 sinon.

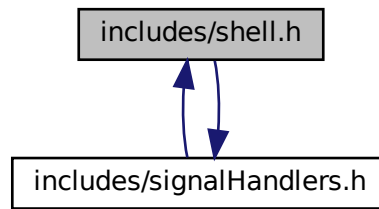
5.5 includes/shell.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include "readcmd.h"
#include "csapp.h"
#include "internCmd.h"
#include "externCmd.h"
#include "jobs.h"
#include "signalHandlers.h"
```

Include dependency graph for shell.h:



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



Variables

- Jobs * jobs

Liste de tous les jobs en cours.

5.5.1 Detailed Description

Author

@OliverMineau

@alexisthach

Version

0.1

Date _____

2023-03-01

Copyright

Copyright (c) 2023

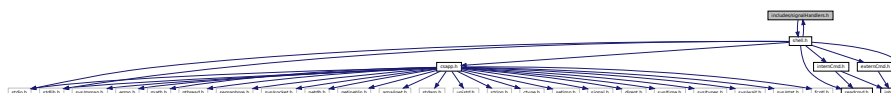
5.6 includes/signalHandlers.h File Reference

```
#include "shell.h"
```

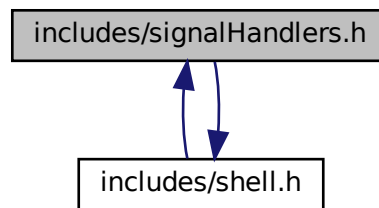
```

// Include dependency graph for signalHandlers.h:

```



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



Functions

- void `childHandler` (int signum)
Tue et supprime les processus terminés.
- void `ctrlCHandler` (int sig)
Intercepte le signal SIGINT.
- void `ctrlZHandler` (int sig)
Intercepte le signal SIGTSTP.

5.6.1 Detailed Description

Author

@OliverMineau
@alexisthach

Version

0.1

Date

2023-03-01

Copyright

Copyright (c) 2023

5.6.2 Function Documentation

5.6.2.1 `childHandler()`

```
void childHandler (  
    int signum )
```

Tue et supprime les processus terminés.

Parameters

<i>signum</i>	
---------------	--

5.6.2.2 ctrlCHandler()

```
void ctrlCHandler (
    int sig )
```

Intercepte le signal SIGINT.

Parameters

<i>sig</i>	
------------	--

5.6.2.3 ctrlZHandler()

```
void ctrlZHandler (
    int sig )
```

Intercepte le signal SIGTSTP.

Parameters

<i>sig</i>	
------------	--

Index

- addJob
 - jobs.h, [18](#)
- childHandler
 - signalHandlers.h, [24](#)
- cmdline, [7](#)
- commandeExterne
 - externCmd.h, [10](#)
- commandeInterne
 - internCmd.h, [13](#)
- ctrlCHandler
 - signalHandlers.h, [25](#)
- ctrlZHandler
 - signalHandlers.h, [25](#)
- deleteJob
 - jobs.h, [18](#)
- externCmd.h
 - commandeExterne, [10](#)
 - seqToChar, [11](#)
- findJobByName
 - jobs.h, [18](#)
- findJobByPID
 - jobs.h, [19](#)
- findJobInFG
 - jobs.h, [19](#)
- findJobNameNum
 - jobs.h, [19](#)
- includes/externCmd.h, [9](#)
- includes/internCmd.h, [11](#)
- includes/jobs.h, [16](#)
- includes/redirection.h, [21](#)
- includes/shell.h, [22](#)
- includes/signalHandlers.h, [23](#)
- internCmd.h
 - commandeInterne, [13](#)
 - isBg, [13](#)
 - isChangeDir, [13](#)
 - isFg, [14](#)
 - isJobs, [14](#)
 - isQuitShell, [15](#)
 - printShellEnv, [15](#)
- isBg
 - internCmd.h, [13](#)
- isChangeDir
 - internCmd.h, [13](#)
- isFg
 - internCmd.h, [14](#)
- isJobs
 - internCmd.h, [14](#)
- isQuitShell
 - internCmd.h, [15](#)
- jobEnFG
 - jobs.h, [20](#)
- Jobs, [7](#)
 - jobs.h, [17](#)
- jobs.h
 - addJob, [18](#)
 - deleteJob, [18](#)
 - findJobByName, [18](#)
 - findJobByPID, [19](#)
 - findJobInFG, [19](#)
 - findJobNameNum, [19](#)
 - jobEnFG, [20](#)
 - Jobs, [17](#)
 - printJobs, [20](#)
- printJobs
 - jobs.h, [20](#)
- printShellEnv
 - internCmd.h, [15](#)
- redirection.h
 - redirectionEntree, [21](#)
 - redirectionSortie, [22](#)
- redirectionEntree
 - redirection.h, [21](#)
- redirectionSortie
 - redirection.h, [22](#)
- rio_t, [8](#)
- seqToChar
 - externCmd.h, [11](#)
- signalHandlers.h
 - childHandler, [24](#)
 - ctrlCHandler, [25](#)
 - ctrlZHandler, [25](#)