Systèmes et Réseaux : Réalisations d'un mini shell

Grand Maxence, Muller Lucie 17/02/2017

Table des matières

1	Partie 1		3
	1.1	Commandes simples	3
	1.2	Commande avec redirections d'entrée ou de sortie	5
	1.3	Séquence de commandes composée de deux commandes reliés par	
		un tube	7
	1.4	Séquence de commandes composée de plusieurs commandes et	
		plusieurs tubes	8
2	Partie Bonus		
	2.1	Exécution de commandes en arrière plan	11
	2.2	Changer l'état du processus au premier plan	11
	2.3	Gestion des zombis	12
	2.4	Commande integrée jobs	13
	2.5	Agir sur les commandes en arrière plan	13

Introduction

Chapitre 1

Partie 1

1.1 Commandes simples

Dans cette partie nous devions implémenté une fonction permettant d'exécuter une commande simple en shell. Pour tester noôtre fonction nous avons tester plusieurs commande n'utilisant ni les pipes ni les redirections dans un shell et dans nôtre programme shell.

Code:

```
void commande_simple(struct cmdline *1){
  int pid = Fork();
  int status;
  if(pid == 0){
    execvp(1->seq[0][0], 1->seq[0]);
    exit(0);
  }
  else{
    waitpid(pid, &status, 0);
  }
}
```

$\begin{array}{c} {\rm Test} \\ {\rm SHELL} \end{array}$

```
maxence@Sybil:/media/maxence/MyLinuxLive1/cours/13_info/s06/SR/
   projetSHELL$ ls
                      rapport.log rapport.tex readcmd.c README.
csapp.c Makefile
   md shell.c
                   test test3 tst
                                        tube_simple.c
csapp.h rapport.aux rapport.pdf rapport.toc readcmd.h
         sujet.pdf test2 text tst.c waitpid1.c
maxence @Sybil:/media/maxence/MyLinuxLive1/cours/l3\_info/s06/SR/
   projetSHELL$ ls −l
total 2688
                                           25 2014 csapp.c
         — 1 maxence maxence
                              20259 \text{ d}\tilde{\text{A}}\odot\text{c}.
        — 1 maxence maxence
                               6105 mars 16
                                               2014 csapp.h
-rwx-
                                               2009 Makefile
-rwx-
      ---- 1 maxence maxence
                                136 oct.
                                           6
         — 1 maxence maxence
                                1499 févr. 8 15:58 rapport.aux
                              30949 févr. 8 15:58 rapport.log
         - 1 maxence maxence
```

```
— 1 maxence maxence 128494 févr. 8 15:58 rapport.pdf
                                   2558 févr. 8 16:02 rapport.tex
933 févr. 8 15:58 rapport.toc
4699 févr. 8 14:07 readcmd.c
-rwx----- 1 maxence maxence
      — 1 maxence maxence
        — 1 maxence maxence
-rwx-
                                  1029 févr. 8 14:02 readcmd.h
14 févr. 7 18:02 README.md
41024 févr. 8 15:45 shell
-rwx----- 1 maxence maxence
      — 1 maxence maxence
-rwx——— 1 maxence maxence
-rwx----- 1 maxence maxence
                                   4338 févr. 8 16:03 shell.c
                                  94332 févr. 7 12:01 sujet
150 févr. 8 13:57 test
-rwx----- 1 maxence maxence
                                                  7 12:01 sujet.pdf
-rwx----- 1 maxence maxence
-rwx----- 1 maxence maxence
                                      3 févr. 8 13:58 test2
-rwx----- 1 maxence maxence
                                      3 févr. 7 15:51 test3
0 févr. 7 15:36 text
-rwx----- 1 maxence maxence
-rwx----- 1 maxence maxence
                                  18008 févr. 8 12:08 tst
-rwx----- 1 maxence maxence
                                    723 févr. 7 13:55 tst.c
-rwx----- 1 maxence maxence
                                    815 janv. 31 15:18 tube_simple.c
                                    657 janv. 20 2009 waitpid1.c
          - 1 maxence maxence
maxence @Sybil:/media/maxence/MyLinuxLive1/cours/l3\_info/s06/SR/
   projetSHELL$ cat test
рр.с
csapp.h
Makefile
readcmd.c
readcmd.h
README, md
shell
shell.c
shell.o
sujet.pdf
test
test2
test3
text
tst
tst.c
tst.o
tube_simple.c
waitpid1.c
```

shell.c

```
shell> ls
csapp.c Makefile
                 rapport.log rapport.tex readcmd.c README.
 md shell.c test test3 tst
                             tube_simple.c
csapp.h rapport.aux rapport.pdf rapport.toc readcmd.h shell
       sujet.pdf test2 text tst.c waitpid1.c
shell> ls -l
total 2688
-rwx—— 1 maxence maxence
                         20259 déc. 25 2014 csapp.c
                         6105 mars 16 2014 csapp.h
       — 1 maxence maxence
-rwx-
-rwx----- 1 maxence maxence
                          136 oct. 6 2009 Makefile
-rwx----- 1 maxence maxence
                          1499 févr. 8 15:58 rapport.aux
                         30949 févr. 8 15:58 rapport.log
-rwx-
     ---- 1 maxence maxence
```

```
1029 févr. 8 14:02 readcmd.h
          - 1 maxence maxence
                                             7 18:02 README.md
                                  14 févr.
-rwx-
        — 1 maxence maxence
             maxence maxence
                               41024 févr.
                                             8 16:06 shell
                               4338 févr. 8 16:03 shell.c
          - 1 maxence maxence
-rwx-
         - 1 maxence maxence
                               94332 févr. 7 12:01 sujet.pdf
-rwx-
         — 1 maxence maxence
                                150 févr.
                                             8 13:57 test
                                             8 13:58 test2
                                  3 févr.
         — 1 maxence maxence
-rwx-
         — 1 maxence maxence
                                   3 févr. 7 15:51 test3
                               - 1 maxence maxence
                                             7 15:36 text
-rwx-
-rwx-
        — 1 maxence maxence
        — 1 maxence maxence
                                 723 févr. 7 13:55 tst.c
                                 815 janv. 31 15:18 tube_simple.c
657 janv. 20 2009 waitpid1.c
        — 1 maxence maxence
-rwx-
      — 1 maxence maxence
shell > cat test
19
pp.c
csapp.h
Makefile
readcmd.c
readcmd.h
README.md
shell
shell.c
shell.o
sujet.pdf
test
test2
test3
t\,e\,x\,t
tst
tst.c
tst.o
tube simple.c
waitpid1.c
       echo "test"
shell >
"test"
shell>
```

1.2 Commande avec redirections d'entrée ou de sortie

Dans cette section nous devions implenter une fonction permettant de gerles redirections d'entrée ou de sortie. Nos test ont suivi les même principes que l'étape précédente à l'exception que seul les commandes avec pipes ne seront pas testées ici.

Code

```
void commande_redirection(struct cmdline *1){
  int fOut = 1; int fIn = 0;

  if(1->in != NULL)
    fIn = open(1->in, O_RDONLY,0);
  if(1->out != NULL)
```

```
fOut = open(1->out, O_WRONLY | O_CREAT, 0700) ;
  int pid = Fork(); int status;
  if(pid == 0){
    if(fOut != 1){
      close(1);
      dup2(fOut, 1);
    }
    if(fIn == 0){
      execvp(1->seq[0][0], 1->seq[0]);
      exit(0);
    }else{
      close(0);
      dup2(fIn, 0);
      execvp(1->seq[0][0], 1->seq[0]);
      exit(0);
  }
  else{
    waitpid(pid, &status, 0);
}
```

Test SHELL

```
maxence@Sybil:/media/maxence/MyLinuxLive1/cours/13 info/s06/SR/
    projetSHELL$ ls -l > test
maxence @Sybil:/media/maxence/MyLinuxLive1/cours/l3\_info/s06/SR/
  projetSHELL$ cat test
total 2688
                                           25 2014 csapp.c
-rwx-
      — 1 maxence maxence
                               20259 \, d\tilde{A}©c.
                               6105 mars 16 2014 csapp.h
        - 1 maxence maxence
-rwx-
                                 136 oct. 6 2009 Makefile
-rwx----- 1 maxence maxence
                                1499 févr. 8 16:10 rapport.aux
-rwx-
      — 1 maxence maxence
                               31866 févr. 8 16:10 rapport.log
      — 1 maxence maxence
-rwx-
      ——— 1 maxence maxence 218712 févr. 8 16:10 rapport.pdf
                                7408 févr. 8 16:16 rapport.tex
933 févr. 8 16:10 rapport.toc
       — 1 maxence maxence
-rwx-
-rwx----- 1 maxence maxence
      ---- 1 maxence maxence
                                4699 févr.
                                              8\ 14:07\ readcmd.c
                                1029 févr. 8 14:02 readcmd.h
14 févr. 7 18:02 README.md
       --- 1 maxence maxence
-rwx-
     — 1 maxence maxence
                               41024 f\tilde{A}©vr. 8 16:06 shell
     — 1 maxence maxence
-rwx-
                               4338 févr. 8 16:03 shell.c
      — 1 maxence maxence
-rwx-
      — 1 maxence maxence
                                             7 12:01 sujet.pdf
                               94332 févr.
                                   0 févr. 8 16:16 test
        — 1 maxence maxence
-rwx-
-rwx-
      ---- 1 maxence maxence
                                   3 févr. 8 13:58 test2
                                   3 févr. 7 15:51 test:
0 févr. 7 15:36 text
                                             7 15:51 test3
      ---- 1 maxence maxence
-rwx-
       — 1 maxence maxence
-rwx-
                               — 1 maxence maxence
      — 1 maxence maxence
-rwx-
                                 815~\mathrm{janv.} 31~15{:}18~\mathrm{tube\_simple.c}
-rwx----- 1 maxence maxence
        — 1 maxence maxence
                                 657 janv. 20 2009 waitpid1.c
maxence@Sybil:/media/maxence/MyLinuxLive1/cours/13_info/s06/SR/
    projetSHELL\$ wc -l < test
23
```

```
shell > ls -l > test
shell> cat test
total 2816
-rwx------ 1 maxence maxence 20259 d	ilde{A}©c. 25 2014 csapp.c
                              6105 mars 16 2014 csapp.h
        — 1 maxence maxence
      — 1 maxence maxence
                               136 oct. 6 2009 Makefile
     ______ 1 maxence maxence 1499 févr. 8 16:10 rapport.aux
______ 1 maxence maxence 31866 févr. 8 16:10 rapport.log
-rwx-
-rwx------ 1 maxence maxence 218712 févr. 8 16:10 rapport.pdf
      -rwx-
-rwx----- 1 maxence maxence
                               4699 févr. 8 14:07 readcmd.c
-rwx----- 1 maxence maxence
                              1029 févr. 8 14:02 readcmd.h
14 févr. 7 18:02 README.md
-rwx----- 1 maxence maxence
                                             7 18:02 README.md
-rwx----- 1 maxence maxence
                              41024 févr. 8 16:23 shell
-rwx----- 1 maxence maxence
                              4343 févr. 8 16:23 shell.c
-rwx----- 1 maxence maxence
                              94332 févr. 7 12:01 suje
1 févr. 8 16:17 test
     — 1 maxence maxence
                                             7 12:01 sujet.pdf
-rwx----- 1 maxence maxence
-rwx----- 1 maxence maxence
                                  1 févr. 8 16:17 test2
                                  3 févr. 7 15:51 test:
0 févr. 7 15:36 text
      — 1 maxence maxence
                                             7 15:51 test3
      — 1 maxence maxence
-rwx-
-rwx----- 1 maxence maxence
                              18008 févr. 8 12:08 tst
                              723 févr.
      — 1 maxence maxence
                                            7 13:55 tst.c
-rwx-
                                815 janv. 31 15:18 tube_simple.c
-rwx----- 1 maxence maxence
       — 1 maxence maxence
                              657 janv. 20 2009 waitpid1.c
shell > wc - l < test
23
shell > wc -c < test
1347
shell > wc - < test > test 2
shell> cat test2
 23 200 1347 -
```

1.3 Séquence de commandes composée de deux commandes reliés par un tube

```
void commande_pipe(struct cmdline *1){
  int tailleSeq; int tmp;
  for(tailleSeq=0; 1->seq[tailleSeq+1]!=0; tailleSeq++);
  int pid = Fork(); int status; int i=0; int desc[2];
  if(pid == 0){
    pipe(desc);
    pid=Fork();
    if(pid != 0){
```

```
dup2(desc[0], 0);
    close(desc[1]);
    while ( (tmp = waitpid(pid, &status, WNOHANG|WUNTRACED)) !=
        pid);
    execvp(l->seq[1][0], l->seq[tailleSeq]);
    close(desc[0]);
    exit(0);
  }else{
    for(i = 1; i<=tailleSeq; i++){</pre>
      if(i+1 \le tailleSeq){
        dup2(desc[1], 1);
        close(desc[0]);
        pipe(desc);
        pid = Fork();
        if(pid != 0){
          dup2(desc[0], 0);
          close(desc[1]);
          waitpid(pid, &status, 0);
          execvp(1->seq[tailleSeq - i][0], 1->seq[tailleSeq-1]);
          exit(0);
        }
      } else{
        dup2(desc[1], 1);
        close(desc[0]);
        execvp(1->seq[0][0], 1->seq[0]);
        exit(0);
    }
 }
}else{
  waitpid(pid, &status, 0);
```

```
shell> ls -l | wc -l
23
shell> ls -l | wc -l | wc -c
3
```

```
maxence@Sybil:/media/maxence/MyLinuxLive1/cours/l3_info/s06/SR/projetSHELL$ ls -l | wc -l
23
maxence@Sybil:/media/maxence/MyLinuxLive1/cours/l3_info/s06/SR/projetSHELL$ ls -l | wc -l | wc -c
3
```

1.4 Séquence de commandes composée de plusieurs commandes et plusieurs tubes

```
void commande1_final(struct cmdline *1){
  if(l->seq[1]!=0){
```

```
int fOut = 1; int fIn = 0;
      if(1->in != NULL)
      fIn = open(1->in, O_RDONLY,0);
    if(1->out != NULL)
      fOut = open(1->out, O_WRONLY | O_CREAT, 0700);
    int tailleSeq; int tmp;
    for(tailleSeq=0; 1->seq[tailleSeq+1]!=0; tailleSeq++);
    int pid = Fork(); int status; int i=0; int desc[2];
    if(pid == 0){
      pipe(desc);
      pid=Fork();
      if(pid != 0){
        dup2(fOut, 1);
        dup2(desc[0], 0);
        close(desc[1]);
        while( (tmp = waitpid(pid, &status, WNOHANG|WUNTRACED)) !=
           pid);
        execvp(l->seq[1][0], l->seq[tailleSeq]);
        close(desc[0]);
        exit(0);
      }else{
        for(i = 1; i<=tailleSeq; i++){</pre>
          if(i+1 <= tailleSeq){</pre>
            dup2(desc[1], 1);
            close(desc[0]);
            pipe(desc);
            pid = Fork();
            if(pid != 0){
              dup2(desc[0], 0);
              close(desc[1]);
              waitpid(pid, &status, 0);
              execvp(l->seq[tailleSeq - i][0], l->seq[tailleSeq-1])
              exit(0);
            }
          } else{
            dup2(fIn, 0);
            dup2(desc[1], 1);
            close(desc[0]);
            execvp(1->seq[0][0], 1->seq[0]);
            close(desc[1]);
            exit(0);
        }
      }
    }else{
      waitpid(pid, &status, 0);
    }
  }else{
   commande_redirection(1);
}
```

```
| maxence@Sybil:/media/maxence/MyLinuxLive1/cours/l3_info/s06/SR/
| projetSHELL$ head -5 < test | tail -2 | sort > tmp2
```

Chapitre 2

Partie Bonus

2.1 Exécution de commandes en arrière plan

```
void commande_bg(struct cmdline *1){
  int pid, status;

pid = Fork();

if(pid == 0){
    execvp(1->seq[0][0], 1->seq[0]);
    exit(0);
}else{
    if(1->bg);
    else{
       waitpid(pid, &status, 0);
    }
}
```

2.2 Changer l'état du processus au premier plan

```
void stop(int sig){
  printf("Fini\n");
  kill(child, SIGKILL);
}
void suspend(int sig){
  printf("Suspendu\n");
  kill(child, SIGSTOP);
}
void commande_signaux(struct cmdline *1){
  int pid = Fork(); int status;
```

```
Signal(SIGINT, stop);
Signal(SIGTSTP, suspend);

if(pid == 0){
    execvp(1->seq[0][0], 1->seq[0]);
    exit(0);
}else{
    child = pid;
    Signal(SIGINT, stop);
    Signal(SIGTSTP, suspend);
    while (waitpid(pid, &status, 0) != pid);
}
```

```
maxence@Sybil:/media/maxence/MyLinuxLive1/cours/l3_info/s06/SR/
projetSHELL$ gedit

C
[2]+ Fini okular
maxence@Sybil:/media/maxence/MyLinuxLive1/cours/l3_info/s06/SR/
projetSHELL$ gedit

Z
[1]+ ArrÃaté gedit
```

```
shell> gedit
^CFini
```

2.3 Gestion des zombis

```
void zombi(int sig)
{
    pid_t pid;

    if ((pid = waitpid(-1, NULL, WNOHANG|WUNTRACED)) < 0)
        unix_error("waitpid error");
    printf("Handler reaped child %d\n", (int)pid);
    return;
}

void commande_zombi(struct cmdline *1){
    Signal( SIGCHLD, zombi);
    int pid = Fork();
    int status;
    if(pid == 0){
        execvp(1->seq[0][0], 1->seq[0]);
        exit(0);
    }
}
```

12

2.4	Commande integrée jobs
2.5	Agir sur les commandes en arrière plan