Compte Rendu TP NACHOS

EUDES Robin, ROSSI Ombeline, BADAMO Romain, MORISON Jack

4 mars 2015

Table des matières

1 Etape 2

1.1 Partie I : Introduction

A l'exécution de putchar.c, on s'attends à avoir le retour suivant :

```
$ ./nachos -x ./putchar
abcd
[...] # retour du syscall halt
```

1.2 Partie II : Entrées-sorties asynchrones

1.2.1 Observation de progtest.cc

```
 ./nachos-userprog -c
test
test # retour de la console
arret si on tape sur q et rien ensuite
arret si on tape sur qMachine halting! # retour de la console
[...]
```

La console se ferme sur la lecture du caractère "q" , et ignore tout ce qu'on a put saisir ensuite. (#Action II.1)

1.2.2 Modifications de progtest.cc

Trace d'exécution :

```
 $ ./nachos-step2 -c
test
<t><e><s><t>


test<t><e><s>< # test suivi de ^D
t> # dernier caractère qui s'affiche après entréé
Machine halting! # ^D en debut de ligne bien pris en compte
```

Après quelques modifications, la terminaison de la console s'effectue sur fin de fichier ou, sur un tty, D̂ en début de ligne. Cependant, nous notons un léger disfonctionement suite à nos modifications : lorsque l'on effectue un D̂ en fin de ligne, le dernier caractère tapé ne sera pas affiché avant que l'on appuie sur entrée. Ce problème se manifeste uniquement dans ce cas, nous pensons à un soucis de buffer. (#Action II.2)

Par ailleurs, on peut observer que chaque caractère est bien encadré de < > . (# Action II.3)

Le test avec un fichier d'entrée et de sortie fonctionne correctement. (#Action II.4)

1.3 Entrées-sorties synchrones

Nous allons maintenant dévelloper une console synchrone. Dans un premier temps, nous copions un squelette fourni. (#Action III.1)

Ensuite, les fonctions de manipulations des caractères sont complétées :

```
 void SynchConsole::SynchPutChar(const char ch)
{
  SemPutChar->P();
  console->PutChar (ch);
  writeDone->P (); // wait for write to finish
  SemPutChar->V();
}

char SynchConsole::SynchGetChar()
{
  SemGetChar->P();
  char ch;
  readAvail->P (); // wait for character to arrive
  ch = console->GetChar ();
  SemGetChar->V();
  return ch;
}
```

La manipulation de String n'est rien de plus qu'une itération d'appels aux fonctions de manipulation des char, après tout, une string n'est qu'une suite de caractères. Explications du fonctionnement sur PutChar :

Dans un premier temps, on prend le sémaphore afin d'assurer qu'un seul caractère est traité à la fois (section critique). L'opération "put" est ensuite effectuée. On attend que le write se termine pour libérer le sémaphore de la section critique, grâce à un second sémaphore (writeDone), qui sera libéré l'écriture effectuée.

On retrouve ce principe sur la fonction getchar.(#Action III.2)