

Nom Prénom :

## CSC3102 – Contrôle Final 2 (1h30) Année 2021-2022

Consignes :

- Répondez aux questions en écrivant directement **dans les cadres fournis** au fil du sujet. **En cas de manque de place**, vous pouvez également écrire en fin de sujet en l'indiquant dans l'encart initial.
- Sont autorisés tous les supports de cours distribués dans le cadre du module ainsi que vos notes personnelles manuscrites.
- La durée est fixée à 1h30.
- Le barème est donné à titre indicatif.

1) [1pt] Donnez la commande qui ajoute les droits d'exécution au script script.sh à n'importe quel utilisateur.

2) [1pt] Donnez la commande qui permet de déplacer un répertoire nommé "Pschit" de votre répertoire de connexion à votre répertoire de travail.

3) [1pt] Donnez la chaîne de commandes (sans structure de contrôle) qui permet de compter le nombre de lignes où apparaît le mot « coucou » dans le fichier toto.txt.

4) [1pt] Donnez la commande qui permet de tuer le processus de PID 34768.

5) [1pt] Donnez la commande qui permet de lister les entrées du répertoire parent qui commencent par 2022.

6) [1pt] Donnez la commande qui permet de trouver dans la sous-arborescence du répertoire /lib les fichiers nommé *aliases.conf*.

7) [6pts] Une bibliothèque propose une collection de livres accessibles en ligne de manière concurrente grâce à un fichier partagé nommé "bibliotheque.txt".

(a) [1pt] Afin d'enregistrer son dernier ouvrage publié, un écrivain **ajoute** son titre **au début** du fichier "bibliotheque.txt". Après avoir fait une copie de bibliotheque.txt, écrivez-y le titre du livre avant d'y ajouter l'ancien contenu. Écrivez le script nouveau\_livre.sh prenant en premier paramètre le titre à inclure.

(b) [2pts] La rentrée littéraire battant son plein, plusieurs écrivains sont amenés à enregistrer au même moment leur publication. Cela pose-t-il un problème de concurrence ? Si oui, **expliquez pourquoi** en illustrant avec un scénario d'ordonnancement menant à un état incohérent et protégez les sections critiques grâce aux scripts P.sh et V.sh utilisés en cours. Sinon, **expliquez pourquoi**.

(c) [1pt] Un lecteur choisit au hasard un livre à lire dans la bibliothèque. Écrivez le script lecteur.sh qui commence par tirer au sort un nombre correspondant à la ligne du livre désigné en utilisant la commande suivante : `n=$(expr $RANDOM % $(wc -l bibliotheque.txt))` , lisez le fichier bibliotheque.txt jusqu'à la n-ième ligne et affichez-la.

(d) [2pts] L'accès concurrent à la bibliothèque par des écrivains et des lecteurs pose-t-il problème ? Si oui, **expliquez pourquoi** en illustrant avec un scénario d'ordonnancement menant à un état incohérent et protégez les sections critiques grâce aux scripts P.sh et V.sh utilisés en cours. Sinon, **expliquez pourquoi**.

nouveau_livre.sh	lecteur.sh
<pre>#!/bin/bash</pre>	<pre>#!/bin/bash  n=\$(expr \$RANDOM % \$(wc -l bibliotheque.txt))</pre>

8) [8pts] Le script comptage.sh compte en continu à partir de 0 suivant une « operation » et un « pas », initialisés au démarrage à + et 1. Tout au long du comptage, l'opération et le pas pourront être modifiés en changeant la variable adéquate sur réception d'une notification émise par un processus externe.

(a) [2pts] Écrivez une première version du script comptage.sh. Pour cela, commencez par déclarer trois variables « operation », « pas » et « n », cette dernière correspondant à la valeur courante dénombrée. Puis, implantez une boucle infinie qui, à chaque seconde, affiche la valeur courante de n avant de la mettre à jour suivant l'opération et le pas voulus.

(b) [2pts] Dans un script inverse.sh, envoyez un signal notifiant au processus comptage.sh (en cours d'exécution) dont le PID est passé en premier paramètre qu'il doit modifier le type de l'opération afin que soit appliquée à l'avenir une soustraction si l'opération courante était une somme et une somme, sinon. Le pas reste inchangé. Complétez le script comptage.sh afin qu'il traite la réception d'un tel signal.

(c) [2pts] Dans un script double.sh, envoyez un signal notifiant au processus comptage.sh (en cours d'exécution) dont le PID est passé en premier paramètre qu'il doit maintenant multiplier par 2 la valeur courante dénombrée. Complétez le script comptage.sh afin qu'il traite la réception d'un tel signal. Pour cela, modifiez le type de l'opération à appliquer ainsi que le pas. Modifiez au besoin le corps de comptage.sh pour prendre en compte cette opération. Attention à protéger le caractère spécial '\*'.

(d) [3pts] Le script pilotage.sh a pour but de traiter les demandes de changement de décompte provenant d'un utilisateur écrivant dans un tube nommé créé à cet effet et ouvert grâce une redirection avancée. Ainsi, après avoir créé et ouvert un tube nommé pilotage.pipe et lancé en tâche de fond un processus comptage.sh, le script pilotage.sh lit en continu sur le tube une demande de l'utilisateur. Celui-ci peut taper « inv » afin d'inverser le signe de l'opération en cours (c'est-à-dire invoquer le script inverse.sh), « double » pour multiplier la valeur courante par deux (c'est-à-dire invoquer le script double.sh) ou « fin » afin de mettre fin au processus comptage.sh s'exécutant en tâche de fond et à pilotage.sh. Toute autre opération n'est pas prise en charge. Écrivez le script pilotage.sh

comptage.sh	inverse.sh
<pre>#!/bin/bash  pas=1 operation=+ n=0  while true ; do     echo \$n      sleep 1 done</pre>	<pre>#!/bin/bash</pre> <hr/> <pre>double.sh</pre> <pre>#!/bin/bash</pre>

```
pilotage.sh
```

```
# ! /bin/bash
```

```
rm -f pilotage.pipe
```

```
while true ; do
```

```
done
```

```
rm pilotage.pipe
```