

INF1070 Utilisation et administration des systèmes informatiques  
Examen Final — Hiver 2025  
Samedi 26 avril — Durée 3 heures  
Ryan Kavanagh et Quentin Stiévenart

- Répondez sur le formulaire (dernière page). Seulement la dernière page sera prise en compte pour la correction.
- Notez aussi votre nom ci-dessous.
- Aucun document ni appareil électronique n'est autorisé.
- L'annexe est détachable.
- Les questions faisant apparaître le symbole ♣ peuvent avoir plusieurs bonnes réponses. Les autres ont une unique bonne réponse.
- Chacune des questions recevra un score entre 0 et 5 points et toutes les questions ont la même pondération.
- Les mauvaises réponses sont pénalisées. Le score d'une question ne peut pas devenir négatif avec les pénalités.
- Le formulaire (dernière page) est lié à votre copie. L'ordre des réponses est différent d'une copie à l'autre. Il ne faut donc pas mélanger le formulaire de réponse de deux copies différentes.

Code permanent :

.....

Prénom :

.....

Nom :

.....

# 1 Expressions régulières

**Question 1 ♣** Les codes postaux canadiens sont des codes alphanumériques formés de six caractères avec le format « LNL NLN », où L est une lettre entre A et Z et N est un chiffre entre 0 et 9. Il y a un espace entre les deux groupes (« LNL », espace, « NLN »). Par exemple, « HOH OH0 » et « P5Q 5P7 » sont des codes postaux, mais « A3B4C6 », « AA5 33B » et « Z5 53H » ne le sont pas. Lesquelles des ERE suivantes ont **une correspondance** avec **au moins** un code postal valide ?

- |  |   |
|--|---|
| <input type="checkbox"/> A (. [0-9] . \$   ^ . [A-Z] . ) | <input type="checkbox"/> D ^ [A-Z] [0-9] [A-Z] [0-9] [A-Z] [0-9] \$ |
| <input type="checkbox"/> B [^0-9] [^A-Z] [^0-9]          | <input type="checkbox"/> E ([A-Z] ? [0-9]) {3}                      |
| <input type="checkbox"/> C ([A-Z] [0-9] ?) {2,}          | <input type="checkbox"/> F (.) (.) (.) \1\2\3                       |

**Question 2 ♣** Une expression régulière  $R_1$  est **équivalente** à une expression régulière  $R_2$  si, pour toute chaîne de caractères  $C$ ,  $R_1$  a une correspondance avec  $C$  si et seulement si  $R_2$  a une correspondance avec  $C$ . Concrètement, cela veut dire que, pour toute entrée, `grep 'R1'` et `grep 'R2'` sélectionneront les mêmes lignes. Indiquez toutes les paires d'expressions régulières étendues (ERE) équivalentes :

- |   |  |
|---|--|
| <input type="checkbox"/> A (a+)? et a*            | <input type="checkbox"/> E a(b[cd]   [bc]d) et a(bc bc)d |
| <input type="checkbox"/> B (a?)+ et a*            | <input type="checkbox"/> F ^ . * \$ et . *               |
| <input type="checkbox"/> C ([ab])[ab] et ([ab])\1 | <input type="checkbox"/> G a?(b c)* et (a?b a?c)*        |
| <input type="checkbox"/> D [^abc\$] et ^ [abc] \$ | <input type="checkbox"/> H a(b c)d et (abd acd)          |

**Question 3 ♣** Chaque ligne du fichier `courriels.html` est un lien vers une URL avec le schéma `mailto`, comme suit :

```
<a href="mailto:alphonse@desjardins.com">Alphonse Desjardins</a>
<a href="mailto:joseph-armand@bombardier.com">Joseph-Armand "Jo" Bombardier</a>
<a href="mailto:mary@la-bolduc.ca" rel="external">Mary Travers</a>
<a href="mailto:paul@manouvellemaison.ca">Paul de Chomedey de Maisonneuve</a>
```

Parmi les commandes suivantes, lesquelles permettent d'extraire de ce fichier toutes les adresses courriels et seulement celles-ci, sans les guillemets ?

- |   |
|---|
| <input type="checkbox"/> A sed -E 's/.*mailto:(.*)".*/\1/' courriels.html           |
| <input type="checkbox"/> B sed -E 's/.*mailto://' courriels.html   sed -E 's/".*//' |
| <input type="checkbox"/> C sed -E 's/".*//' courriels.html   sed -E 's/.*://'       |
| <input type="checkbox"/> D sed -E 's/.*mailto:(["+).*/\1/' courriels.html           |

**Question 4 ♣** Lesquelles des URLs suivantes ont une correspondance avec l'ERE « ((f|ht)tps?://|mailto:)(infor?|ma|thé)+(tiques?)?.uqam.ca » ?

- |   |   |
|---|---|
| <input type="checkbox"/> A ftp://info.uqam.ca           | <input type="checkbox"/> E mailto:thé@uqam.ca           |
| <input type="checkbox"/> B mailto:thétiques?.uqam.ca    | <input type="checkbox"/> F ://ma.uqam.ca                |
| <input type="checkbox"/> C http://mathématiques.uqam.ca | <input type="checkbox"/> G https://uqam.ca              |
| <input type="checkbox"/> D HTTP://MA.UQAM.CA            | <input type="checkbox"/> H https://informatique.uqam.ca |

**Question 5 ♣** Lesquelles des ERE suivantes ont une correspondance avec la chaîne « épaisissait » ?

- |   |   |
|---|---|
| <input type="checkbox"/> A (.)\1\1      | <input type="checkbox"/> E ((..)\2      |
| <input type="checkbox"/> B (..)(..)\2\1 | <input type="checkbox"/> F (..)*\1      |
| <input type="checkbox"/> C ^ (.) . * \1 | <input type="checkbox"/> G ((..)\1      |
| <input type="checkbox"/> D (is) . + \1  | <input type="checkbox"/> H (..)(..)\2\1 |

## 2 Administration

**Question 6 ♣** Parmi les éléments suivants, lesquels font partie des tâches d'un gestionnaire de paquets ?

- A Compter les paquets réseau
- B Désinstaller un paquet
- C Installer des fichiers exécutables
- D Installer les dépendances d'un paquet
- E Surveiller en temps réel l'utilisation CPU et RAM des logiciels installés

**Question 7 ♣** Parmi les énoncés suivants, lesquels sont forcément vrais pour un démon (daemon) ?

- A Il peut être lancé automatiquement au démarrage du système
- B Il s'exécute en arrière plan
- C Il s'exécute en avant plan et interagit directement avec l'utilisateur
- D Il est automatiquement fermé dès que l'utilisateur ferme son terminal
- E Il s'exécute avec une priorité élevée

**Question 8 ♣** Parmi les énoncés suivants, lesquels sont vrais ?

- A Les fichiers de journalisation peuvent être généralement analysés avec des outils comme **grep** pour rechercher des événements spécifiques
- B Les journaux des services sont généralement stockés dans `/etc/log/`
- C Les journaux des services sont généralement stockés dans le dossier utilisateur
- D Certains fichiers de journalisation peuvent être automatiquement archivés
- E Les fichiers de journalisation contiennent des nouvelles importantes sur les nouveaux paquets logiciels disponibles
- F Les journaux des services sont généralement stockés dans `/var/log/`

### 3 Processus

Les 4 questions suivantes se font dans le contexte suivant.

```
$ ps -efj
UID      PID  PPID  PGID   SID  C  STIME TTY      TIME CMD
root      1    0     1     1  0  11:33 ?        00:00:00 /sbin/init
root     507    1   507   507  0  11:33 ?        00:00:00 /usr/sbin/anacron -d -q -s
avahi    508    1   508   508  0  11:33 ?        00:00:00 avahi-daemon: running [debian.local]
root     509    1   509   509  0  11:33 ?        00:00:00 /usr/sbin/cron -f
message+ 510    1   510   510  0  11:33 ?        00:00:00 /usr/bin/dbus-daemon --system --address=sys
root     512    1   512   512  0  11:33 ?        00:00:00 /usr/libexec/low-memory-monitor
debian  1845  1768  1845  1768  0  11:34 pts/0    00:00:00 tmux
debian  1847    1  1847  1847  0  11:34 ?        00:00:00 tmux
debian  1848  1847  1848  1848  0  11:34 pts/1    00:00:00 -bash
debian  1858  1848  1858  1848  0  11:35 pts/1    00:00:00 sleep 1200
debian  1859  1847  1859  1859  0  11:35 pts/2    00:00:00 -bash
debian  1865  1859  1865  1859  1  11:35 pts/2    00:00:06 htop
debian  1866  1847  1866  1866  0  11:35 pts/3    00:00:00 -bash
debian  1951  1847  1951  1951  0  11:41 pts/4    00:00:00 -bash
debian  1957  1951  1957  1951  0  11:41 pts/4    00:00:00 vim
debian  1959  1957  1957  1951  0  11:41 pts/4    00:00:00 cat
debian  1960  1866  1960  1866  0  11:41 pts/3    00:00:00 ps -efj
```

**Question 9 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |  |
|--|--|
| <input type="checkbox"/> A L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code>    | <input type="checkbox"/> D L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>sleep</code>        |
| <input type="checkbox"/> B L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>sleep</code>        | <input type="checkbox"/> E L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code> |
| <input type="checkbox"/> C L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>avahi-daemon</code> | <input type="checkbox"/> F L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>sleep</code>       |

**Question 10** Parmi les affirmations suivantes, laquelle est vraie pour le contexte donné ?

- |  |   |
|--|---|
| <input type="checkbox"/> A L'identifiant de processus de la commande <code>sleep</code> est <code>11:35</code> | <input type="checkbox"/> C L'identifiant de processus de la commande <code>sleep</code> est <code>1848</code> |
| <input type="checkbox"/> B L'identifiant de processus de la commande <code>sleep</code> est <code>pts/1</code> | <input type="checkbox"/> D L'identifiant de processus de la commande <code>sleep</code> est <code>1858</code> |

**Question 11 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |   |  |
|---|--|
| <input type="checkbox"/> A La commande <code>cat</code> a été lancée directement depuis <code>vim</code>      | <input type="checkbox"/> D la commande <code>ps -efj</code> a été lancée directement depuis <code>vim</code> |
| <input type="checkbox"/> B la commande <code>cat</code> a été lancée directement depuis <code>bash</code>     | <input type="checkbox"/> E la commande <code>cat</code> a été lancée directement depuis <code>tmux</code>    |
| <input type="checkbox"/> C la commande <code>ps -efj</code> a été lancée directement depuis <code>bash</code> |  |

Pour les questions suivantes, on a le contexte suivant.

```
$ jobs -l
[1] 3532 Running gnome-calculator &
[2] 3643 Running nautilus &
[3] 3710 Running xeyes &
```

**Question 12 ♣** Après avoir effectué avec succès la commande `kill 3643`, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>gnome-calculator</code> est terminé             | <input type="checkbox"/> F Le processus <code>xeyes</code> s'exécute en arrière-plan            |
| <input type="checkbox"/> B Le processus <code>gnome-calculator</code> s'exécute en avant-plan | <input type="checkbox"/> G Le processus <code>gnome-calculator</code> s'exécute en arrière-plan |
| <input type="checkbox"/> C Le processus <code>nautilus</code> s'exécute en avant-plan         | <input type="checkbox"/> H Le processus <code>nautilus</code> est terminé                       |
| <input type="checkbox"/> D Le processus <code>xeyes</code> est terminé                        | <input type="checkbox"/> I Le processus <code>xeyes</code> s'exécute en avant-plan              |
| <input type="checkbox"/> E Le processus <code>nautilus</code> s'exécute en arrière-plan       |   |

**Question 13 ♣** À la place de la commande de la question précédente, si j'avais effectué la commande `fg %3` et ensuite entré `^C` (Contrôle-C) au clavier, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>xeyes</code> s'exécute en arrière-plan            | <input type="checkbox"/> E Le processus <code>gnome-calculator</code> s'exécute en avant-plan |
| <input type="checkbox"/> B Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> F Le processus <code>nautilus</code> s'exécute en avant-plan         |
| <input type="checkbox"/> C Le processus <code>xeyes</code> est terminé                          | <input type="checkbox"/> G Le processus <code>nautilus</code> est terminé                     |
| <input type="checkbox"/> D Le processus <code>gnome-calculator</code> est terminé               | <input type="checkbox"/> H Le processus <code>xeyes</code> s'exécute en avant-plan            |
|   | <input type="checkbox"/> I Le processus <code>nautilus</code> s'exécute en arrière-plan       |

## 4 Shell avancé

**Question 14** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
x="a"
y="b"
z="c d"
echo "$x" '$y' $z
```

- |                                      |                                      |  |                                      |
|--------------------------------------|--------------------------------------|--|--------------------------------------|
| <input type="checkbox"/> A a \$y c d | <input type="checkbox"/> B a \$y c d | <input type="checkbox"/> C \$x \$y \$z | <input type="checkbox"/> D \$x y c d |
|--------------------------------------|--------------------------------------|--|--------------------------------------|

**Question 15 ♣** Soit le script shell suivant :

```
#!/bin/sh

x=9
x="9*$x"
echo "$x"

y=0
y="$(echo "$y")"
echo "$y"

z=1
z="$(echo '$z ')"
echo "$z"
```

Quelles lignes apparaissent, entre autres, dans sa sortie ?

- A 0       B 81       C \$x       D \$z       E 9\*9       F \$y       G 1

**Question 16 ♣** Le répertoire courant contient le fichier « Mon rapport.txt ». Vous avez tous les droits d'accès sur le répertoire courant et ses fichiers. Lesquels des énoncés suivants sont vrais ?

- A La commande « a="Mon rapport.txt"; rm \$a » supprime le fichier « Mon rapport.txt »  
 B La commande « a="Mon\*.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 C La commande « a=Mon\ rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 D La commande « a=Mon rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 E La commande « a="Mon rapport.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 F La commande « a="Mon"; rm "\$a\*" » supprime le fichier « Mon rapport.txt »

**Question 17** Le script shell suivant :

```
#!/bin/sh

cd /bin
pwd > /tmp/pwd.log
( cd /etc; pwd; )
pwd >> /tmp/pwd.log
{ cd /usr; pwd; }
pwd >> /tmp/pwd.log
{ cd /tmp; pwd; } | rev
pwd >> /tmp/pwd.log
```

a la sortie suivante :

```
/etc
/usr
pmt/
```

Quelle sera le contenu du fichier « /tmp/pwd.log » ?

- A /bin       B /bin       C /bin       D /bin       E /bin       F /bin       G /bin  
           /bin            /etc            /bin            /bin            /bin            /bin            /etc  
           /usr            /usr            /usr            /bin            /bin            /bin            /etc  
           /usr            /usr            /tmp            /tmp            /usr            /bin            /tmp

**Question 18 ♣** La commande « `cmd` » présente dans le `PATH` a toujours le même comportement et produit la sortie suivante dans le terminal :

```
abc
xbz
```

Parmi les énoncés suivants, lesquels sont forcément vrais ?

- A Si « `cmd > /dev/null` » n'affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.
- B Il est possible que la conduite « `cmd | grep q` » affiche la ligne « `abc` » dans le terminal.
- C Si « `cmd 2> /dev/null` » n'affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.
- D Si « `cmd | grep q` » n'affiche aucune ligne, alors « `cmd > /dev/null` » n'affiche rien dans le terminal.

**Question 19 ♣** Vous avez tous les droits d'accès sur le répertoire courant qui est vide. La commande « `oui` » produit toujours le texte « `pomme` » sur sa sortie standard. La commande « `non` » produit toujours le texte « `pomme` » sur sa sortie d'erreur. Suites à quelles commandes le fichier « `fruits` » contiendra-t-il le mot « `pomme` » ?

- |  |   |   |
|--|---|---|
| <input type="checkbox"/> A <code>non 2&gt;&amp;1 &gt; fruits</code>  | <input type="checkbox"/> C <code>oui &gt; fruits &gt;&amp;2</code>  | <input type="checkbox"/> E <code>oui &gt;&amp;2 2&gt; fruits</code> |
| <input type="checkbox"/> B <code>non 2&gt; fruits 1&gt;&amp;2</code> | <input type="checkbox"/> D <code>non &gt; fruits 2&gt;&amp;1</code> | <input type="checkbox"/> F <code>oui &gt; fruits 2&gt;&amp;1</code> |

**Question 20 ♣** Soit les commandes suivantes, où chaque commande est identifiée par un commentaire :

```
sleep 1; sleep 1           # C1
sleep 1 || sleep 2        # C2
sleep 1 | sleep 2         # C3
{ sleep 2; sleep 2; } | sleep 2 # C4
sleep 2 && sleep 2         # C5
sleep 3 & sleep 4 &       # C6
```

Lesquels des énoncés suivants sont vrais ?

- |  |  |
|--|--|
| <input type="checkbox"/> A La commande C5 dure environ 4 secondes. | <input type="checkbox"/> E La commande C4 dure environ 4 secondes. |
| <input type="checkbox"/> B La commande C3 dure environ 2 secondes. | <input type="checkbox"/> F La commande C6 dure environ 7 secondes. |
| <input type="checkbox"/> C La commande C2 dure environ 3 seconde.  | <input type="checkbox"/> G La commande C1 dure environ 2 secondes. |
| <input type="checkbox"/> D La commande C2 dure environ 2 seconde.  | <input type="checkbox"/> H La commande C4 dure environ 6 secondes. |

**Question 21 ♣** La sortie de « `echo "$PATH"` » est « `/usr/bin:/bin` ». La sortie de « `type rm` » est « `rm est /usr/bin/rm` ». « `/usr/bin/rm` » et « `/bin/sh` » sont conforme au standard POSIX. Vous avez tous les droits d'accès sur « `/a` » et « `/a/rm` » et le script « `/a/rm` » a le contenu suivant :

```
#!/bin/sh
echo "rm"
```

Lesquels des énoncés suivants sont vrais ?

- A La commande « `/bin/sh /a/rm` » affiche exactement le texte « `rm` ».
- B La commande « `/a/rm rm` » affiche exactement le texte « `rm` ».
- C La commande « `cd /a; rm rm` » affiche exactement le texte « `rm` ».
- D La commande « `rm rm` » affiche exactement le texte « `rm` ».
- E La commande « `PATH="/a:/usr/bin:/bin" rm rm` » affiche exactement le texte « `rm` ».
- F La commande « `export PATH="/a"; rm` » affiche exactement le texte « `rm` ».
- G La commande « `PATH="/a/rm" rm rm` » affiche exactement le texte « `rm` ».

**Question 22 ♣** Parmi les commandes suivantes, lesquelles produisent systématiquement la sortie suivante ?

```
abc
def
```

- |                            |                                |                            |                               |
|----------------------------|--------------------------------|----------------------------|-------------------------------|
| <input type="checkbox"/> A | echo abc   tac; echo def   tac | <input type="checkbox"/> D | ( echo def; echo abc; )   tac |
| <input type="checkbox"/> B | echo def; echo abc;   tac      | <input type="checkbox"/> E | { echo def; echo abc; }   tac |
| <input type="checkbox"/> C | echo def   echo abc   tac      | <input type="checkbox"/> F | { echo def; echo abc; } > tac |

**Question 23 ♣** Vous avez tous les droits d'accès sur le répertoire « /final » et son fichier « /final/banane ». Lesquels des énoncés suivants sont vrais ?

- |                            |  |                            |  |
|----------------------------|--|----------------------------|--|
| <input type="checkbox"/> A | « rm /final/banane    echo "banane" » affiche le texte « banane ».           | <input type="checkbox"/> D | « ( exit 1 ) && echo "banane" » affiche le texte « banane ».                 |
| <input type="checkbox"/> B | « ( exit 0 )    echo "banane" » affiche le texte « banane ».                 | <input type="checkbox"/> E | « rm /final/banane && echo "banane" » supprime le fichier « /final/banane ». |
| <input type="checkbox"/> C | « rm /final/banane    echo "banane" » supprime le fichier « /final/banane ». | <input type="checkbox"/> F | « rm /final/banane && echo "banane" » affiche le texte « banane ».           |

## 5 Scriptage

**Question 24** Soit la mise en situation suivante. Quelle sera la sortie produite par « ./test.sh » à la fin ?

```
rak@koios:~$ ls -l
total 5
-rwxrwxr-x 1 rak rak 134 2025-04-13 20:18 test.sh
rak@koios:~$ cat test.sh
#!/bin/sh

if echo "test" | grep t* > /dev/null; then
    echo "1"
else
    echo "2"
fi

if echo "abcd" | grep t* > /dev/null; then
    echo "3"
else
    echo "4"
fi
rak@koios:~$ ./test.sh
```

- |                            |        |                            |        |                            |        |                            |        |
|----------------------------|--------|----------------------------|--------|----------------------------|--------|----------------------------|--------|
| <input type="checkbox"/> A | 2<br>3 | <input type="checkbox"/> B | 2<br>4 | <input type="checkbox"/> C | 1<br>3 | <input type="checkbox"/> D | 1<br>4 |
|----------------------------|--------|----------------------------|--------|----------------------------|--------|----------------------------|--------|

**Question 25** Soit le script « ./mystere » suivant :

```
#!/bin/bash
x=0
y=3
f() {
    local x=$((x + 1))
    y=$((y - 1))
}
for a in "$@"; do f; done
echo "x:$x y:$y"
```

Quelle sera la sortie de « ./mystere a b c » ?

A x:0 y:3

B x:3 y:3

C x:0 y:0

D x:3 y:0

**Question 26** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
c=""
for f in pomme banane ananas; do
    if [ "$f" = "ananas" ]; then
        exit 1
    fi
    c="$f $c"
done
echo "$c"
```

A pomme banane ananas

B 1

C ananas banane pomme

D Aucune sortie

**Question 27 ♣** Vous avez toutes les permissions sur le répertoire courant. Le script « ./map » a les permissions 555 et le contenu suivant :

```
#!/bin/sh
F="$1"
shift
for x in "$@"; do
    $F "$x"
done
```

Parmi les énoncés suivants, lesquels sont vrais ?

A « ./map echo "Bonjour INF1070" » produit une ligne, avec seulement « Bonjour ».

B « ./map head \* » produit, entre autres, la ligne « #!/bin/sh ».

C « ./map "echo Bonjour INF1070" » produit, entre autres, la ligne « Bonjour ».

D « ./map echo "Bonjour INF1070" » produit la ligne « Bonjour INF1070 ».

E « ./map echo a b c » produit, entre autres, la ligne « a ».

**Question 28** Soient le script shell « `./script1` » suivant :

```
#!/bin/sh
exit 123
```

et le script shell « `./script2` » suivant :

```
#!/bin/sh
if ./script1; then
    echo "$?"
else
    echo "$?"
    echo "$?"
fi
```

Les deux fichiers sont exécutable. Quelle sera la sortie de « `./script2` » ?

- |                                       |  |                                |
|---------------------------------------|--|--------------------------------|
| <input type="checkbox"/> A 0          | <input type="checkbox"/> C Aucune sortie | <input type="checkbox"/> E 123 |
| <input type="checkbox"/> B 123<br>123 | <input type="checkbox"/> D 123<br>0      |                                |

## 6 Réseau

**Question 29** Quel protocole se charge de router des paquets de données entre interfaces réseaux de machines connectées à internet ?

- |  |  |
|--|--|
| <input type="checkbox"/> A Le protocole SSH  | <input type="checkbox"/> D Le protocole DNS  |
| <input type="checkbox"/> B Le protocole HTTP | <input type="checkbox"/> E Le protocole ADDR |
| <input type="checkbox"/> C Le protocole DHCP | <input type="checkbox"/> F Le protocole IP   |

**Question 30** Si je visite le site `https://inf1070.uqam.ca`, Firefox m'affiche un cadenas à côté de l'URL dans le navigateur. Cela signifie que :

- |  |  |
|--|--|
| <input type="checkbox"/> A La connexion s'est faite au travers du protocole DNS  | <input type="checkbox"/> D La page n'est pas modifiable par l'utilisateur                    |
| <input type="checkbox"/> B La connexion est sécurisée au travers d'un certificat | <input type="checkbox"/> E Il faut entrer un mot de passe pour accéder au contenu de la page |
| <input type="checkbox"/> C La page ne sera jamais modifiée                       |  |

## Annexe détachable

**bg** [*job\_id*] Exécute une tâche en arrière plan

**cat** Concaténer des fichiers et les afficher sur la sortie standard. Si aucun fichier n'est fourni, ou si le fichier est « - », **cat** lit l'entrée standard.

**cd** [*rep*] Change le répertoire de travail en cours par *rep*.

### Descripteurs de fichiers

1 = sortie standard, 2 = sortie standard d'erreur

**echo** Affiche une ligne de texte.

**fg** [*job\_id*] Exécute une tâche en avant plan

**exit** Termine le shell avec un code de retour. Si omis, le code de retour est celui de la dernière commande exécutée.

**for** NOM in MOTS ...; do COMMANDES; done

Exécute des commandes pour chaque membre d'une liste. Pour chaque élément dans MOTS, NOM est défini à cet élément, et les COMMANDES sont exécutées. Le code de sortie est celui la dernière commande exécutée.

**grep** Recherche les lignes correspondant à un motif donné. Par défaut, **grep** affiche les lignes qui contiennent une correspondance au motif. Code de retour : 0 si une ligne a été trouvée, 1 sinon.

**head** Afficher le début du contenu de fichiers.

**if** Exécute des commandes selon une condition.

Usage : **if** *lst1*; **then** *lst2*; [ **else** *lst3*; ] **fi**

La liste « *lst1* » est exécutée. Si elle se termine avec le code zéro, alors la liste « *lst2* » est exécutée. Sinon, la liste « *lst3* » est exécutée si elle existe. Le code de retour de l'ensemble est celui de la dernière commande exécutée ou zéro si aucune condition n'était vraie.

**jobs** Affiche les tâches qui sont exécutées.

-1 : donne plus d'information sur les tâches, incluant le numéro de la tâche et son PID

**kill** Envoie un signal à des processus. Les processus sont

identifié par leurs PID ou par des spécifications de tâches (préfixées par %). Si on ne précise pas de signal, TERM est envoyé, qui demande au processus de se terminer.

**ls** Afficher le contenu de répertoires.

-1 : utiliser un format d'affichage long.

**PATH** Variable utilisée par le shell

Elle contient la liste ordonnée des répertoires où le shell va chercher ses commandes avoir vérifié ses primitives.

### Paramètres

**\$#** : le nombre de paramètres positionnels.

**\$@** : l'ensemble des paramètres positionnels sous forme de liste.

**ps** Affiche un instantané des processus en cours. Options :

-e : affiche tous les processus

-f : liste plus de colonnes

-j : liste encore plus de colonnes

**pwd** Afficher le nom du répertoire de travail actuel.

**rm** Effacer des fichiers et des répertoires. Code de retour : 0 en cas de succès, 1 en cas d'échec.

**sed** Éditeur de flux pour filtrer et transformer du texte.

-E utilise des expressions régulières étendues.

**shift** Décale la liste des arguments d'une position vers la gauche.

**sleep** Attendre une durée déterminée (en secondes).

**tac** Concatène et affiche des fichiers à l'envers (la dernière ligne en premier).

**test**, [

Teste l'expression entre crochets [ ] et renvoie un état de sortie correspondant au résultat de la comparaison (0 pour vrai, 1 pour faux).

**arg1 = arg2** : Teste l'égalité entre deux chaînes de caractères.





Remplissez complètement la case (■) pour indiquer une bonne réponse (pas de ×, pas de ✓, pas de cercles).  
Les réponses aux questions sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte.

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

← Codez les 8 chiffres de votre code permanent ci-contre, et inscrivez-le à nouveau ci-dessous avec vos nom et prénom. **Les copies sans code permanent ne seront pas corrigées.**

Code permanent :
.....
Prénom :
.....
Nom :
.....

- Question 1 :  A  B  C  D  E  F
- Question 2 :  A  B  C  D  E  F  G  H
- Question 3 :  A  B  C  D
- Question 4 :  A  B  C  D  E  F  G  H
- Question 5 :  A  B  C  D  E  F  G  H
- Question 6 :  A  B  C  D  E
- Question 7 :  A  B  C  D  E
- Question 8 :  A  B  C  D  E  F
- Question 9 :  A  B  C  D  E  F
- Question 10 :  A  B  C  D
- Question 11 :  A  B  C  D  E
- Question 12 :  A  B  C  D  E  F  G  H  I
- Question 13 :  A  B  C  D  E  F  G  H  I
- Question 14 :  A  B  C  D
- Question 15 :  A  B  C  D  E  F  G

- Question 16 :  A  B  C  D  E  F
- Question 17 :  A  B  C  D  E  F  G
- Question 18 :  A  B  C  D
- Question 19 :  A  B  C  D  E  F
- Question 20 :  A  B  C  D  E  F  G  H
- Question 21 :  A  B  C  D  E  F  G
- Question 22 :  A  B  C  D  E  F
- Question 23 :  A  B  C  D  E  F
- Question 24 :  A  B  C  D
- Question 25 :  A  B  C  D
- Question 26 :  A  B  C  D
- Question 27 :  A  B  C  D  E
- Question 28 :  A  B  C  D  E
- Question 29 :  A  B  C  D  E  F
- Question 30 :  A  B  C  D  E



INF1070 Utilisation et administration des systèmes informatiques  
Examen Final — Hiver 2025  
Samedi 26 avril — Durée 3 heures  
Ryan Kavanagh et Quentin Stiévenart

- Répondez sur le formulaire (dernière page). Seulement la dernière page sera prise en compte pour la correction.
- Notez aussi votre nom ci-dessous.
- Aucun document ni appareil électronique n'est autorisé.
- L'annexe est détachable.
- Les questions faisant apparaître le symbole ♣ peuvent avoir plusieurs bonnes réponses. Les autres ont une unique bonne réponse.
- Chacune des questions recevra un score entre 0 et 5 points et toutes les questions ont la même pondération.
- Les mauvaises réponses sont pénalisées. Le score d'une question ne peut pas devenir négatif avec les pénalités.
- Le formulaire (dernière page) est lié à votre copie. L'ordre des réponses est différent d'une copie à l'autre. Il ne faut donc pas mélanger le formulaire de réponse de deux copies différentes.

Code permanent :

.....

Prénom :

.....

Nom :

.....

# 1 Expressions régulières

**Question 1 ♣** Les codes postaux canadiens sont des codes alphanumériques formés de six caractères avec le format « LNL NLN », où L est une lettre entre A et Z et N est un chiffre entre 0 et 9. Il y a un espace entre les deux groupes (« LNL », espace, « NLN »). Par exemple, « HOH OH0 » et « P5Q 5P7 » sont des codes postaux, mais « A3B4C6 », « AA5 33B » et « Z5 53H » ne le sont pas. Lesquelles des ERE suivantes ont **une correspondance** avec **au moins** un code postal valide ?

- |  |   |
|--|---|
| <input type="checkbox"/> A (. [0-9] . \$ ^ . [A-Z] . ) | <input type="checkbox"/> D ([A-Z] ?[0-9]){3}                      |
| <input type="checkbox"/> B [^0-9] [^A-Z] [^0-9]        | <input type="checkbox"/> E ^[A-Z] [0-9] [A-Z] [0-9] [A-Z] [0-9]\$ |
| <input type="checkbox"/> C ([A-Z] [0-9] ?){2,}         | <input type="checkbox"/> F (.) (.) (.) \1\2\3                     |

**Question 2 ♣** Une expression régulière  $R_1$  est **équivalente** à une expression régulière  $R_2$  si, pour toute chaîne de caractères  $C$ ,  $R_1$  a une correspondance avec  $C$  si et seulement si  $R_2$  a une correspondance avec  $C$ . Concrètement, cela veut dire que, pour toute entrée, `grep 'R1'` et `grep 'R2'` sélectionneront les mêmes lignes. Indiquez toutes les paires d'expressions régulières étendues (ERE) équivalentes :

- |  |   |
|--|---|
| <input type="checkbox"/> A a(b[cd]   [bc]d) et a(bc bc)d | <input type="checkbox"/> E [^abc\$] et ^[abc]\$   |
| <input type="checkbox"/> B a?(b c)* et (a?b a?c)*        | <input type="checkbox"/> F ([ab])[ab] et ([ab])\1 |
| <input type="checkbox"/> C a(b c)d et (abd acd)          | <input type="checkbox"/> G (a+)? et a*            |
| <input type="checkbox"/> D ^.*\$ et .*                   | <input type="checkbox"/> H (a?)+ et a*            |

**Question 3 ♣** Chaque ligne du fichier `courriels.html` est un lien vers une URL avec le schéma `mailto`, comme suit :

```
<a href="mailto:alphonse@desjardins.com">Alphonse Desjardins</a>
<a href="mailto:joseph-armand@bombardier.com">Joseph-Armand "Jo" Bombardier</a>
<a href="mailto:mary@la-bolduc.ca" rel="external">Mary Travers</a>
<a href="mailto:paul@manouvellemaison.ca">Paul de Chomedey de Maisonneuve</a>
```

Parmi les commandes suivantes, lesquelles permettent d'extraire de ce fichier toutes les adresses courriels et seulement celles-ci, sans les guillemets ?

- |   |
|---|
| <input type="checkbox"/> A sed -E 's/.*mailto:(["]+).*/\1/' courriels.html          |
| <input type="checkbox"/> B sed -E 's/".*//\' courriels.html   sed -E 's/.*:/'       |
| <input type="checkbox"/> C sed -E 's/.*mailto:(.*)".*/\1/' courriels.html           |
| <input type="checkbox"/> D sed -E 's/.*mailto:/' courriels.html   sed -E 's/".*//\' |

**Question 4 ♣** Lesquelles des URLs suivantes ont une correspondance avec l'ERE « ((f|ht)tps?://|mailto:)(infor?|ma|thé)+(tiques?)?.uqam.ca » ?

- |   |  |
|---|--|
| <input type="checkbox"/> A ftp://info.uqam.ca           | <input type="checkbox"/> E mailto:thé@uqam.ca        |
| <input type="checkbox"/> B https://informatique.uqam.ca | <input type="checkbox"/> F ://ma.uqam.ca             |
| <input type="checkbox"/> C https://uqam.ca              | <input type="checkbox"/> G mailto:thétiques?.uqam.ca |
| <input type="checkbox"/> D http://mathématiques.uqam.ca | <input type="checkbox"/> H HTTP://MA.UQAM.CA         |

**Question 5 ♣** Lesquelles des ERE suivantes ont une correspondance avec la chaîne « épaisissait » ?

- |                                     |   |
|-------------------------------------|---|
| <input type="checkbox"/> A (..)*\1  | <input type="checkbox"/> E ((..)\1      |
| <input type="checkbox"/> B ((..)\2  | <input type="checkbox"/> F (..)(..)\2\1 |
| <input type="checkbox"/> C ^(..)*\1 | <input type="checkbox"/> G (..)(..)\2\1 |
| <input type="checkbox"/> D (is).+\1 | <input type="checkbox"/> H (.)\1\1      |

## 2 Administration

**Question 6 ♣** Parmi les éléments suivants, lesquels font partie des tâches d'un gestionnaire de paquets ?

- A Installer les dépendances d'un paquet
- B Surveiller en temps réel l'utilisation CPU et RAM des logiciels installés
- C Compter les paquets réseau
- D Installer des fichiers exécutables
- E Désinstaller un paquet

**Question 7 ♣** Parmi les énoncés suivants, lesquels sont forcément vrais pour un démon (daemon) ?

- A Il s'exécute en avant plan et interagit directement avec l'utilisateur
- B Il s'exécute en arrière plan
- C Il peut être lancé automatiquement au démarrage du système
- D Il s'exécute avec une priorité élevée
- E Il est automatiquement fermé dès que l'utilisateur ferme son terminal

**Question 8 ♣** Parmi les énoncés suivants, lesquels sont vrais ?

- A Les journaux des services sont généralement stockés dans le dossier utilisateur
- B Les fichiers de journalisation contiennent des nouvelles importantes sur les nouveaux paquets logiciels disponibles
- C Les journaux des services sont généralement stockés dans `/var/log/`
- D Les journaux des services sont généralement stockés dans `/etc/log/`
- E Les fichiers de journalisation peuvent être généralement analysés avec des outils comme `grep` pour rechercher des événements spécifiques
- F Certains fichiers de journalisation peuvent être automatiquement archivés

### 3 Processus

Les 4 questions suivantes se font dans le contexte suivant.

```
$ ps -efj
UID      PID  PPID  PGID   SID  C  STIME TTY      TIME CMD
root      1    0     1     1  0  11:33 ?        00:00:00 /sbin/init
root     507    1   507   507  0  11:33 ?        00:00:00 /usr/sbin/anacron -d -q -s
avahi    508    1   508   508  0  11:33 ?        00:00:00 avahi-daemon: running [debian.local]
root     509    1   509   509  0  11:33 ?        00:00:00 /usr/sbin/cron -f
message+ 510    1   510   510  0  11:33 ?        00:00:00 /usr/bin/dbus-daemon --system --address=sys
root     512    1   512   512  0  11:33 ?        00:00:00 /usr/libexec/low-memory-monitor
debian  1845  1768  1845  1768  0  11:34 pts/0    00:00:00 tmux
debian  1847    1  1847  1847  0  11:34 ?        00:00:00 tmux
debian  1848  1847  1848  1848  0  11:34 pts/1    00:00:00 -bash
debian  1858  1848  1858  1848  0  11:35 pts/1    00:00:00 sleep 1200
debian  1859  1847  1859  1859  0  11:35 pts/2    00:00:00 -bash
debian  1865  1859  1865  1859  1  11:35 pts/2    00:00:06 htop
debian  1866  1847  1866  1866  0  11:35 pts/3    00:00:00 -bash
debian  1951  1847  1951  1951  0  11:41 pts/4    00:00:00 -bash
debian  1957  1951  1957  1951  0  11:41 pts/4    00:00:00 vim
debian  1959  1957  1957  1951  0  11:41 pts/4    00:00:00 cat
debian  1960  1866  1960  1866  0  11:41 pts/3    00:00:00 ps -efj
```

**Question 9 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |   |
|--|---|
| <input type="checkbox"/> A L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>avahi-daemon</code> | <input type="checkbox"/> D L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>sleep</code>       |
| <input type="checkbox"/> B L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code>   | <input type="checkbox"/> E L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code> |
| <input type="checkbox"/> C L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>sleep</code>         | <input type="checkbox"/> F L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>sleep</code>     |

**Question 10** Parmi les affirmations suivantes, laquelle est vraie pour le contexte donné ?

- |   |   |
|---|---|
| <input type="checkbox"/> A L'identifiant de processus de la commande <code>sleep</code> est 1858  | <input type="checkbox"/> C L'identifiant de processus de la commande <code>sleep</code> est 1848  |
| <input type="checkbox"/> B L'identifiant de processus de la commande <code>sleep</code> est pts/1 | <input type="checkbox"/> D L'identifiant de processus de la commande <code>sleep</code> est 11:35 |

**Question 11 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |   |
|--|---|
| <input type="checkbox"/> A la commande <code>cat</code> a été lancée directement depuis <code>tmux</code>    | <input type="checkbox"/> D la commande <code>cat</code> a été lancée directement depuis <code>bash</code>     |
| <input type="checkbox"/> B la commande <code>ps -efj</code> a été lancée directement depuis <code>vim</code> | <input type="checkbox"/> E la commande <code>ps -efj</code> a été lancée directement depuis <code>bash</code> |
| <input type="checkbox"/> C La commande <code>cat</code> a été lancée directement depuis <code>vim</code>     |   |

Pour les questions suivantes, on a le contexte suivant.

```
$ jobs -l
[1] 3532 Running gnome-calculator &
[2] 3643 Running nautilus &
[3] 3710 Running xeyes &
```

**Question 12 ♣** Après avoir effectué avec succès la commande `kill 3643`, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>nautilus</code> s'exécute en avant-plan           | <input type="checkbox"/> F Le processus <code>nautilus</code> s'exécute en arrière-plan       |
| <input type="checkbox"/> B Le processus <code>xeyes</code> s'exécute en avant-plan              | <input type="checkbox"/> G Le processus <code>xeyes</code> est terminé                        |
| <input type="checkbox"/> C Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> H Le processus <code>gnome-calculator</code> s'exécute en avant-plan |
| <input type="checkbox"/> D Le processus <code>nautilus</code> est terminé                       | <input type="checkbox"/> I Le processus <code>gnome-calculator</code> est terminé             |
| <input type="checkbox"/> E Le processus <code>xeyes</code> s'exécute en arrière-plan            |   |

**Question 13 ♣** À la place de la commande de la question précédente, si j'avais effectué la commande `fg %3` et ensuite entré `^C` (Contrôle-C) au clavier, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>xeyes</code> s'exécute en arrière-plan            | <input type="checkbox"/> E Le processus <code>gnome-calculator</code> s'exécute en avant-plan |
| <input type="checkbox"/> B Le processus <code>xeyes</code> s'exécute en avant-plan              | <input type="checkbox"/> F Le processus <code>xeyes</code> est terminé                        |
| <input type="checkbox"/> C Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> G Le processus <code>nautilus</code> est terminé                     |
| <input type="checkbox"/> D Le processus <code>nautilus</code> s'exécute en avant-plan           | <input type="checkbox"/> H Le processus <code>gnome-calculator</code> est terminé             |
|   | <input type="checkbox"/> I Le processus <code>nautilus</code> s'exécute en arrière-plan       |

## 4 Shell avancé

**Question 14** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
x="a"
y="b"
z="c      d"
echo "$x" '$y' $z
```

- |  |   |                                      |                                      |
|--|---|--------------------------------------|--------------------------------------|
| <input type="checkbox"/> A \$x \$y \$z | <input type="checkbox"/> B a \$y c      d | <input type="checkbox"/> C \$x y c d | <input type="checkbox"/> D a \$y c d |
|--|---|--------------------------------------|--------------------------------------|

**Question 15 ♣** Soit le script shell suivant :

```
#!/bin/sh
```

```
x=9
x="9*$x"
echo "$x"
```

```
y=0
y="$(echo "$y")"
echo "$y"
```

```
z=1
z="$(echo '$z ')"
echo "$z"
```

Quelles lignes apparaissent, entre autres, dans sa sortie ?

- A \$x       B 9\*9       C 0       D \$z       E \$y       F 1       G 81

**Question 16 ♣** Le répertoire courant contient le fichier « Mon rapport.txt ». Vous avez tous les droits d'accès sur le répertoire courant et ses fichiers. Lesquels des énoncés suivants sont vrais ?

- A La commande « a="Mon rapport.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 B La commande « a="Mon\*.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 C La commande « a="Mon"; rm "\$a\*" » supprime le fichier « Mon rapport.txt »  
 D La commande « a=Mon rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 E La commande « a=Mon\ rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 F La commande « a="Mon rapport.txt"; rm \$a » supprime le fichier « Mon rapport.txt »

**Question 17** Le script shell suivant :

```
#!/bin/sh
```

```
cd /bin
pwd > /tmp/pwd.log
( cd /etc; pwd; )
pwd >> /tmp/pwd.log
{ cd /usr; pwd; }
pwd >> /tmp/pwd.log
{ cd /tmp; pwd; } | rev
pwd >> /tmp/pwd.log
```

a la sortie suivante :

```
/etc
/usr
pmt/
```

Quelle sera le contenu du fichier « /tmp/pwd.log » ?

- A /bin       B /bin       C /bin       D /bin       E /bin       F /bin       G /bin  
           /bin            /etc            /bin            /bin            /bin            /bin            /etc  
           /usr            /etc            /bin            /bin            /usr            /bin            /usr  
           /usr            /tmp            /bin            /usr            /tmp            /tmp            /usr

**Question 18 ♣** La commande « `cmd` » présente dans le `PATH` a toujours le même comportement et produit la sortie suivante dans le terminal :

```
abc
xbz
```

Parmis les énoncés suivants, lesquels sont forcément vrais ?

- A Il est possible que la conduite « `cmd | grep q` » affiche la ligne « `abc` » dans le terminal.
- B Si « `cmd 2> /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.
- C Si « `cmd | grep q` » n’affiche aucune ligne, alors « `cmd > /dev/null` » n’affiche rien dans le terminal.
- D Si « `cmd > /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.

**Question 19 ♣** Vous avez tous les droits d’accès sur le répertoire courant qui est vide. La commande « `oui` » produit toujours le texte « `pomme` » sur sa sortie standard. La commande « `non` » produit toujours le texte « `pomme` » sur sa sortie d’erreur. Suites à quelles commandes le fichier « `fruits` » contiendra-t-il le mot « `pomme` » ?

- |   |  |   |
|---|--|---|
| <input type="checkbox"/> A <code>oui &gt;&amp;2 2&gt; fruits</code> | <input type="checkbox"/> C <code>non 2&gt; fruits 1&gt;&amp;2</code> | <input type="checkbox"/> E <code>non 2&gt;&amp;1 &gt; fruits</code> |
| <input type="checkbox"/> B <code>oui &gt; fruits &gt;&amp;2</code>  | <input type="checkbox"/> D <code>oui &gt; fruits 2&gt;&amp;1</code>  | <input type="checkbox"/> F <code>non &gt; fruits 2&gt;&amp;1</code> |

**Question 20 ♣** Soit les commandes suivantes, où chaque commande est identifiée par un commentaire :

```
sleep 1; sleep 1           # C1
sleep 1 || sleep 2        # C2
sleep 1 | sleep 2         # C3
{ sleep 2; sleep 2; } | sleep 2 # C4
sleep 2 && sleep 2         # C5
sleep 3 & sleep 4 &       # C6
```

Lesquels des énoncés suivants sont vrais ?

- |  |  |
|--|--|
| <input type="checkbox"/> A La commande C3 dure environ 2 secondes. | <input type="checkbox"/> E La commande C2 dure environ 2 seconde.  |
| <input type="checkbox"/> B La commande C4 dure environ 4 secondes. | <input type="checkbox"/> F La commande C4 dure environ 6 secondes. |
| <input type="checkbox"/> C La commande C2 dure environ 3 seconde.  | <input type="checkbox"/> G La commande C1 dure environ 2 secondes. |
| <input type="checkbox"/> D La commande C5 dure environ 4 secondes. | <input type="checkbox"/> H La commande C6 dure environ 7 secondes. |

**Question 21 ♣** La sortie de « `echo "$PATH"` » est « `/usr/bin:/bin` ». La sortie de « `type rm` » est « `rm est /usr/bin/rm` ». « `/usr/bin/rm` » et « `/bin/sh` » sont conforme au standard POSIX. Vous avez tous les droits d’accès sur « `/a` » et « `/a/rm` » et le script « `/a/rm` » a le contenu suivant :

```
#!/bin/sh
echo "rm"
```

Lesquels des énoncés suivants sont vrais ?

- A La commande « `export PATH="/a"; rm` » affiche exactement le texte « `rm` ».
- B La commande « `PATH="/a:/usr/bin:/bin" rm rm` » affiche exactement le texte « `rm` ».
- C La commande « `PATH="/a/rm" rm rm` » affiche exactement le texte « `rm` ».
- D La commande « `/bin/sh /a/rm` » affiche exactement le texte « `rm` ».
- E La commande « `cd /a; rm rm` » affiche exactement le texte « `rm` ».
- F La commande « `rm rm` » affiche exactement le texte « `rm` ».
- G La commande « `/a/rm rm` » affiche exactement le texte « `rm` ».

**Question 22 ♣** Parmi les commandes suivantes, lesquelles produisent systématiquement la sortie suivante ?

```
abc
def
```

- |                            |                                |                            |                               |
|----------------------------|--------------------------------|----------------------------|-------------------------------|
| <input type="checkbox"/> A | echo def; echo abc;   tac      | <input type="checkbox"/> D | { echo def; echo abc; }   tac |
| <input type="checkbox"/> B | echo def   echo abc   tac      | <input type="checkbox"/> E | ( echo def; echo abc; )   tac |
| <input type="checkbox"/> C | echo abc   tac; echo def   tac | <input type="checkbox"/> F | { echo def; echo abc; } > tac |

**Question 23 ♣** Vous avez tous les droits d'accès sur le répertoire « /final » et son fichier « /final/banane ». Lesquels des énoncés suivants sont vrais ?

- |                            |  |                            |  |
|----------------------------|--|----------------------------|--|
| <input type="checkbox"/> A | « rm /final/banane && echo "banane" » affiche le texte « banane ».           | <input type="checkbox"/> D | « rm /final/banane && echo "banane" » supprime le fichier « /final/banane ». |
| <input type="checkbox"/> B | « ( exit 0 )    echo "banane" » affiche le texte « banane ».                 | <input type="checkbox"/> E | « ( exit 1 ) && echo "banane" » affiche le texte « banane ».                 |
| <input type="checkbox"/> C | « rm /final/banane    echo "banane" » supprime le fichier « /final/banane ». | <input type="checkbox"/> F | « rm /final/banane    echo "banane" » affiche le texte « banane ».           |

## 5 Scriptage

**Question 24** Soit la mise en situation suivante. Quelle sera la sortie produite par « ./test.sh » à la fin ?

```
rak@koios:~$ ls -l
total 5
-rwxrwxr-x 1 rak rak 134 2025-04-13 20:18 test.sh
rak@koios:~$ cat test.sh
#!/bin/sh

if echo "test" | grep t* > /dev/null; then
    echo "1"
else
    echo "2"
fi

if echo "abcd" | grep t* > /dev/null; then
    echo "3"
else
    echo "4"
fi
rak@koios:~$ ./test.sh
```

- |                            |        |                            |        |                            |        |                            |        |
|----------------------------|--------|----------------------------|--------|----------------------------|--------|----------------------------|--------|
| <input type="checkbox"/> A | 2<br>4 | <input type="checkbox"/> B | 1<br>3 | <input type="checkbox"/> C | 2<br>3 | <input type="checkbox"/> D | 1<br>4 |
|----------------------------|--------|----------------------------|--------|----------------------------|--------|----------------------------|--------|

**Question 25** Soit le script « ./mystere » suivant :

```
#!/bin/bash
x=0
y=3
f() {
    local x=$((x + 1))
    y=$((y - 1))
}
for a in "$@"; do f; done
echo "x:$x y:$y"
```

Quelle sera la sortie de « ./mystere a b c » ?

- A x:0 y:3                       B x:3 y:3                       C x:3 y:0                       D x:0 y:0

**Question 26** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
c=""
for f in pomme banane ananas; do
    if [ "$f" = "ananas" ]; then
        exit 1
    fi
    c="$f $c"
done
echo "$c"
```

- A pomme banane ananas                       C 1  
 B ananas banane pomme                       D Aucune sortie

**Question 27 ♣** Vous avez toutes les permissions sur le répertoire courant. Le script « ./map » a les permissions 555 et le contenu suivant :

```
#!/bin/sh
F="$1"
shift
for x in "$@"; do
    $F "$x"
done
```

Parmi les énoncés suivants, lesquels sont vrais ?

- A « ./map echo "Bonjour INF1070" » produit la ligne « Bonjour INF1070 ».  
 B « ./map head \* » produit, entre autres, la ligne « #!/bin/sh ».  
 C « ./map echo a b c » produit, entre autres, la ligne « a ».  
 D « ./map echo "Bonjour INF1070" » produit une ligne, avec seulement « Bonjour ».  
 E « ./map "echo Bonjour INF1070" » produit, entre autres, la ligne « Bonjour ».

**Question 28** Soient le script shell « `./script1` » suivant :

```
#!/bin/sh
exit 123
```

et le script shell « `./script2` » suivant :

```
#!/bin/sh
if ./script1; then
    echo "$?"
else
    echo "$?"
    echo "$?"
fi
```

Les deux fichiers sont exécutable. Quelle sera la sortie de « `./script2` » ?

- |                                |                                |  |                              |
|--------------------------------|--------------------------------|--|------------------------------|
| <input type="checkbox"/> A 123 | <input type="checkbox"/> B 123 | <input type="checkbox"/> C Aucune sortie | <input type="checkbox"/> E 0 |
| 123                            | 0                              | <input type="checkbox"/> D 123           |                              |

## 6 Réseau

**Question 29** Quel protocole se charge de router des paquets de données entre interfaces réseaux de machines connectées à internet ?

- |   |  |
|---|--|
| <input type="checkbox"/> A Le protocole SSH | <input type="checkbox"/> D Le protocole DHCP |
| <input type="checkbox"/> B Le protocole IP  | <input type="checkbox"/> E Le protocole HTTP |
| <input type="checkbox"/> C Le protocole DNS | <input type="checkbox"/> F Le protocole ADDR |

**Question 30** Si je visite le site `https://inf1070.uqam.ca`, Firefox m'affiche un cadenas à côté de l'URL dans le navigateur. Cela signifie que :

- |  |  |
|--|--|
| <input type="checkbox"/> A Il faut entrer un mot de passe pour accéder au contenu de la page | <input type="checkbox"/> C La connexion s'est faite au travers du protocole DNS  |
| <input type="checkbox"/> B La page ne sera jamais modifiée                                   | <input type="checkbox"/> D La page n'est pas modifiable par l'utilisateur        |
|  | <input type="checkbox"/> E La connexion est sécurisée au travers d'un certificat |

## Annexe détachable

**bg** [*job\_id*] Exécute une tâche en arrière plan

**cat** Concaténer des fichiers et les afficher sur la sortie standard. Si aucun fichier n'est fourni, ou si le fichier est « - », **cat** lit l'entrée standard.

**cd** [*rep*] Change le répertoire de travail en cours par *rep*.

### Descripteurs de fichiers

1 = sortie standard, 2 = sortie standard d'erreur

**echo** Affiche une ligne de texte.

**fg** [*job\_id*] Exécute une tâche en avant plan

**exit** Termine le shell avec un code de retour. Si omis, le code de retour est celui de la dernière commande exécutée.

**for** NOM in MOTS ...; do COMMANDES; done

Exécute des commandes pour chaque membre d'une liste. Pour chaque élément dans MOTS, NOM est défini à cet élément, et les COMMANDES sont exécutées. Le code de sortie est celui la dernière commande exécutée.

**grep** Recherche les lignes correspondant à un motif donné. Par défaut, **grep** affiche les lignes qui contiennent une correspondance au motif. Code de retour : 0 si une ligne a été trouvée, 1 sinon.

**head** Afficher le début du contenu de fichiers.

**if** Exécute des commandes selon une condition.

Usage : **if** *lst1*; **then** *lst2*; [ **else** *lst3*; ] **fi**

La liste « *lst1* » est exécutée. Si elle se termine avec le code zéro, alors la liste « *lst2* » est exécutée. Sinon, la liste « *lst3* » est exécutée si elle existe. Le code de retour de l'ensemble est celui de la dernière commande exécutée ou zéro si aucune condition n'était vraie.

**jobs** Affiche les tâches qui sont exécutées.

-1 : donne plus d'information sur les tâches, incluant le numéro de la tâche et son PID

**kill** Envoie un signal à des processus. Les processus sont

identifié par leurs PID ou par des spécifications de tâches (préfixées par %). Si on ne précise pas de signal, TERM est envoyé, qui demande au processus de se terminer.

**ls** Afficher le contenu de répertoires.

-1 : utiliser un format d'affichage long.

**PATH** Variable utilisée par le shell

Elle contient la liste ordonnée des répertoires où le shell va chercher ses commandes avoir vérifié ses primitives.

### Paramètres

**\$#** : le nombre de paramètres positionnels.

**\$@** : l'ensemble des paramètres positionnels sous forme de liste.

**ps** Affiche un instantané des processus en cours. Options :

-e : affiche tous les processus

-f : liste plus de colonnes

-j : liste encore plus de colonnes

**pwd** Afficher le nom du répertoire de travail actuel.

**rm** Effacer des fichiers et des répertoires. Code de retour : 0 en cas de succès, 1 en cas d'échec.

**sed** Éditeur de flux pour filtrer et transformer du texte.

-E utilise des expressions régulières étendues.

**shift** Décale la liste des arguments d'une position vers la gauche.

**sleep** Attendre une durée déterminée (en secondes).

**tac** Concatène et affiche des fichiers à l'envers (la dernière ligne en premier).

**test**, [

Teste l'expression entre crochets [ ] et renvoie un état de sortie correspondant au résultat de la comparaison (0 pour vrai, 1 pour faux).

**arg1 = arg2** : Teste l'égalité entre deux chaînes de caractères.





Remplissez complètement la case (■) pour indiquer une bonne réponse (pas de ×, pas de ✓, pas de cercles).  
Les réponses aux questions sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte.

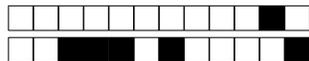
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

← Codez les 8 chiffres de votre code permanent ci-contre, et inscrivez-le à nouveau ci-dessous avec vos nom et prénom. **Les copies sans code permanent ne seront pas corrigées.**

Code permanent :
.....
Prénom :
.....
Nom :
.....

- Question 1 :  A  B  C  D  E  F
- Question 2 :  A  B  C  D  E  F  G  H
- Question 3 :  A  B  C  D
- Question 4 :  A  B  C  D  E  F  G  H
- Question 5 :  A  B  C  D  E  F  G  H
- Question 6 :  A  B  C  D  E
- Question 7 :  A  B  C  D  E
- Question 8 :  A  B  C  D  E  F
- Question 9 :  A  B  C  D  E  F
- Question 10 :  A  B  C  D
- Question 11 :  A  B  C  D  E
- Question 12 :  A  B  C  D  E  F  G  H  I
- Question 13 :  A  B  C  D  E  F  G  H  I
- Question 14 :  A  B  C  D
- Question 15 :  A  B  C  D  E  F  G

- Question 16 :  A  B  C  D  E  F
- Question 17 :  A  B  C  D  E  F  G
- Question 18 :  A  B  C  D
- Question 19 :  A  B  C  D  E  F
- Question 20 :  A  B  C  D  E  F  G  H
- Question 21 :  A  B  C  D  E  F  G
- Question 22 :  A  B  C  D  E  F
- Question 23 :  A  B  C  D  E  F
- Question 24 :  A  B  C  D
- Question 25 :  A  B  C  D
- Question 26 :  A  B  C  D
- Question 27 :  A  B  C  D  E
- Question 28 :  A  B  C  D  E
- Question 29 :  A  B  C  D  E  F
- Question 30 :  A  B  C  D  E



+2/14/33+

INF1070 Utilisation et administration des systèmes informatiques  
Examen Final — Hiver 2025  
Samedi 26 avril — Durée 3 heures  
Ryan Kavanagh et Quentin Stiévenart

- Répondez sur le formulaire (dernière page). Seulement la dernière page sera prise en compte pour la correction.
- Notez aussi votre nom ci-dessous.
- Aucun document ni appareil électronique n'est autorisé.
- L'annexe est détachable.
- Les questions faisant apparaître le symbole ♣ peuvent avoir plusieurs bonnes réponses. Les autres ont une unique bonne réponse.
- Chacune des questions recevra un score entre 0 et 5 points et toutes les questions ont la même pondération.
- Les mauvaises réponses sont pénalisées. Le score d'une question ne peut pas devenir négatif avec les pénalités.
- Le formulaire (dernière page) est lié à votre copie. L'ordre des réponses est différent d'une copie à l'autre. Il ne faut donc pas mélanger le formulaire de réponse de deux copies différentes.

Code permanent :

.....

Prénom :

.....

Nom :

.....

# 1 Expressions régulières

**Question 1 ♣** Les codes postaux canadiens sont des codes alphanumériques formés de six caractères avec le format « LNL NLN », où L est une lettre entre A et Z et N est un chiffre entre 0 et 9. Il y a un espace entre les deux groupes (« LNL », espace, « NLN »). Par exemple, « HOH OH0 » et « P5Q 5P7 » sont des codes postaux, mais « A3B4C6 », « AA5 33B » et « Z5 53H » ne le sont pas. Lesquelles des ERE suivantes ont **une correspondance** avec **au moins** un code postal valide ?

- |   |  |
|---|--|
| <input type="checkbox"/> A <code>(.)(.)(.) \1\2\3</code>                  | <input type="checkbox"/> D <code>([A-Z][0-9] ?){2,}</code> |
| <input type="checkbox"/> B <code>^[A-Z][0-9][A-Z][0-9][A-Z][0-9]\$</code> | <input type="checkbox"/> E <code>([A-Z] ?[0-9]){3}</code>  |
| <input type="checkbox"/> C <code>(.[0-9].\$ ^.[A-Z].)</code>              | <input type="checkbox"/> F <code>[^0-9][^A-Z][^0-9]</code> |

**Question 2 ♣** Une expression régulière  $R_1$  est **équivalente** à une expression régulière  $R_2$  si, pour toute chaîne de caractères  $C$ ,  $R_1$  a une correspondance avec  $C$  si et seulement si  $R_2$  a une correspondance avec  $C$ . Concrètement, cela veut dire que, pour toute entrée, `grep 'R1'` et `grep 'R2'` sélectionneront les mêmes lignes. Indiquez toutes les paires d'expressions régulières étendues (ERE) équivalentes :

- |   |   |
|---|---|
| <input type="checkbox"/> A <code>(a?)+</code> et <code>a*</code>                  | <input type="checkbox"/> E <code>a?(b c)*</code> et <code>(a?b a?c)*</code> |
| <input type="checkbox"/> B <code>(a+)?</code> et <code>a*</code>                  | <input type="checkbox"/> F <code>([ab])[ab]</code> et <code>([ab])\1</code> |
| <input type="checkbox"/> C <code>a(b c)d</code> et <code>(abd acd)</code>         | <input type="checkbox"/> G <code>[^abc\$]</code> et <code>^[abc]\$</code>   |
| <input type="checkbox"/> D <code>a(b[cd]  [bc]d)</code> et <code>a(bc bc)d</code> | <input type="checkbox"/> H <code>^.*\$</code> et <code>.*</code>            |

**Question 3 ♣** Chaque ligne du fichier `courriels.html` est un lien vers une URL avec le schéma `mailto`, comme suit :

```
<a href="mailto:alphonse@desjardins.com">Alphonse Desjardins</a>
<a href="mailto:joseph-armand@bombardier.com">Joseph-Armand "Jo" Bombardier</a>
<a href="mailto:mary@la-bolduc.ca" rel="external">Mary Travers</a>
<a href="mailto:paul@manouvellemaison.ca">Paul de Chomedey de Maisonneuve</a>
```

Parmi les commandes suivantes, lesquelles permettent d'extraire de ce fichier toutes les adresses courriels et seulement celles-ci, sans les guillemets ?

- |   |
|---|
| <input type="checkbox"/> A <code>sed -E 's/".*//' courriels.html   sed -E 's/.*:/'</code>       |
| <input type="checkbox"/> B <code>sed -E 's/.*mailto:(.*)".*/\1/' courriels.html</code>          |
| <input type="checkbox"/> C <code>sed -E 's/.*mailto:/' courriels.html   sed -E 's/".*//'</code> |
| <input type="checkbox"/> D <code>sed -E 's/.*mailto:(["+).*/\1/' courriels.html</code>          |

**Question 4 ♣** Lesquelles des URLs suivantes ont une correspondance avec l'ERE « `((f|ht)tps?://|mailto:)(infor?|ma|thé)+(tiques?)?.uqam.ca` » ?

- |  |  |
|--|--|
| <input type="checkbox"/> A <code>http://mathématiques.uqam.ca</code> | <input type="checkbox"/> E <code>ftp://info.uqam.ca</code>           |
| <input type="checkbox"/> B <code>mailto:thétiques?.uqam.ca</code>    | <input type="checkbox"/> F <code>https://informatique.uqam.ca</code> |
| <input type="checkbox"/> C <code>HTTP://MA.UQAM.CA</code>            | <input type="checkbox"/> G <code>https://uqam.ca</code>              |
| <input type="checkbox"/> D <code>mailto:thé@uqam.ca</code>           | <input type="checkbox"/> H <code>://ma.uqam.ca</code>                |

**Question 5 ♣** Lesquelles des ERE suivantes ont une correspondance avec la chaîne « `épaississait` » ?

- |  |  |
|--|--|
| <input type="checkbox"/> A <code>((..)\1</code>  | <input type="checkbox"/> E <code>(..)(..)\2\1</code> |
| <input type="checkbox"/> B <code>^(..)*\1</code> | <input type="checkbox"/> F <code>(..)(..)\2\1</code> |
| <input type="checkbox"/> C <code>((..)\2</code>  | <input type="checkbox"/> G <code>(..)*\1</code>      |
| <input type="checkbox"/> D <code>(.)\1\1</code>  | <input type="checkbox"/> H <code>(is).+\1</code>     |

## 2 Administration

**Question 6 ♣** Parmi les éléments suivants, lesquels font partie des tâches d'un gestionnaire de paquets ?

- A Installer les dépendances d'un paquet
- B Compter les paquets réseau
- C Surveiller en temps réel l'utilisation CPU et RAM des logiciels installés
- D Installer des fichiers exécutables
- E Désinstaller un paquet

**Question 7 ♣** Parmi les énoncés suivants, lesquels sont forcément vrais pour un démon (daemon) ?

- A Il peut être lancé automatiquement au démarrage du système
- B Il s'exécute en arrière plan
- C Il s'exécute en avant plan et interagit directement avec l'utilisateur
- D Il est automatiquement fermé dès que l'utilisateur ferme son terminal
- E Il s'exécute avec une priorité élevée

**Question 8 ♣** Parmi les énoncés suivants, lesquels sont vrais ?

- A Les journaux des services sont généralement stockés dans `/etc/log/`
- B Les fichiers de journalisation peuvent être généralement analysés avec des outils comme `grep` pour rechercher des événements spécifiques
- C Les journaux des services sont généralement stockés dans le dossier utilisateur
- D Les journaux des services sont généralement stockés dans `/var/log/`
- E Certains fichiers de journalisation peuvent être automatiquement archivés
- F Les fichiers de journalisation contiennent des nouvelles importantes sur les nouveaux paquets logiciels disponibles

### 3 Processus

Les 4 questions suivantes se font dans le contexte suivant.

```
$ ps -efj
UID      PID  PPID  PGID   SID  C  STIME TTY      TIME CMD
root      1    0     1     1  0  11:33 ?        00:00:00 /sbin/init
root     507    1   507   507  0  11:33 ?        00:00:00 /usr/sbin/anacron -d -q -s
avahi    508    1   508   508  0  11:33 ?        00:00:00 avahi-daemon: running [debian.local]
root     509    1   509   509  0  11:33 ?        00:00:00 /usr/sbin/cron -f
message+ 510    1   510   510  0  11:33 ?        00:00:00 /usr/bin/dbus-daemon --system --address=sys
root     512    1   512   512  0  11:33 ?        00:00:00 /usr/libexec/low-memory-monitor
debian  1845  1768  1845  1768  0  11:34 pts/0    00:00:00 tmux
debian  1847    1  1847  1847  0  11:34 ?        00:00:00 tmux
debian  1848  1847  1848  1848  0  11:34 pts/1    00:00:00 -bash
debian  1858  1848  1858  1848  0  11:35 pts/1    00:00:00 sleep 1200
debian  1859  1847  1859  1859  0  11:35 pts/2    00:00:00 -bash
debian  1865  1859  1865  1859  1  11:35 pts/2    00:00:06 htop
debian  1866  1847  1866  1866  0  11:35 pts/3    00:00:00 -bash
debian  1951  1847  1951  1951  0  11:41 pts/4    00:00:00 -bash
debian  1957  1951  1957  1951  0  11:41 pts/4    00:00:00 vim
debian  1959  1957  1957  1951  0  11:41 pts/4    00:00:00 cat
debian  1960  1866  1960  1866  0  11:41 pts/3    00:00:00 ps -efj
```

**Question 9 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |  |
|--|--|
| <input type="checkbox"/> A L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code>  | <input type="checkbox"/> D L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>sleep</code>        |
| <input type="checkbox"/> B L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>sleep</code>       | <input type="checkbox"/> E L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>avahi-daemon</code> |
| <input type="checkbox"/> C L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code> | <input type="checkbox"/> F L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>sleep</code>          |

**Question 10** Parmi les affirmations suivantes, laquelle est vraie pour le contexte donné ?

- |  |  |
|--|--|
| <input type="checkbox"/> A L'identifiant de processus de la commande <code>sleep</code> est <code>pts/1</code> | <input type="checkbox"/> C L'identifiant de processus de la commande <code>sleep</code> est <code>1848</code>  |
| <input type="checkbox"/> B L'identifiant de processus de la commande <code>sleep</code> est <code>1858</code>  | <input type="checkbox"/> D L'identifiant de processus de la commande <code>sleep</code> est <code>11:35</code> |

**Question 11 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |   |
|--|---|
| <input type="checkbox"/> A la commande <code>ps -efj</code> a été lancée directement depuis <code>vim</code> | <input type="checkbox"/> D la commande <code>ps -efj</code> a été lancée directement depuis <code>bash</code> |
| <input type="checkbox"/> B la commande <code>cat</code> a été lancée directement depuis <code>tmux</code>    | <input type="checkbox"/> E La commande <code>cat</code> a été lancée directement depuis <code>vim</code>      |
| <input type="checkbox"/> C la commande <code>cat</code> a été lancée directement depuis <code>bash</code>    |   |

Pour les questions suivantes, on a le contexte suivant.

```
$ jobs -l
[1] 3532 Running gnome-calculator &
[2] 3643 Running nautilus &
[3] 3710 Running xeyes &
```

**Question 12 ♣** Après avoir effectué avec succès la commande `kill 3643`, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>nautilus</code> s'exécute en arrière-plan         | <input type="checkbox"/> E Le processus <code>nautilus</code> s'exécute en avant-plan |
| <input type="checkbox"/> B Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> F Le processus <code>xeyes</code> est terminé                |
| <input type="checkbox"/> C Le processus <code>gnome-calculator</code> s'exécute en avant-plan   | <input type="checkbox"/> G Le processus <code>gnome-calculator</code> est terminé     |
| <input type="checkbox"/> D Le processus <code>nautilus</code> est terminé                       | <input type="checkbox"/> H Le processus <code>xeyes</code> s'exécute en avant-plan    |
|   | <input type="checkbox"/> I Le processus <code>xeyes</code> s'exécute en arrière-plan  |

**Question 13 ♣** À la place de la commande de la question précédente, si j'avais effectué la commande `fg %3` et ensuite entré `^C` (Contrôle-C) au clavier, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>nautilus</code> s'exécute en arrière-plan       | <input type="checkbox"/> E Le processus <code>gnome-calculator</code> s'exécute en arrière-plan |
| <input type="checkbox"/> B Le processus <code>xeyes</code> est terminé                        | <input type="checkbox"/> F Le processus <code>gnome-calculator</code> est terminé               |
| <input type="checkbox"/> C Le processus <code>gnome-calculator</code> s'exécute en avant-plan | <input type="checkbox"/> G Le processus <code>nautilus</code> s'exécute en avant-plan           |
| <input type="checkbox"/> D Le processus <code>nautilus</code> est terminé                     | <input type="checkbox"/> H Le processus <code>xeyes</code> s'exécute en avant-plan              |
|   | <input type="checkbox"/> I Le processus <code>xeyes</code> s'exécute en arrière-plan            |

## 4 Shell avancé

**Question 14** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
x="a"
y="b"
z="c d"
echo "$x" '$y' $z
```

- |                                      |                                      |                                      |  |
|--------------------------------------|--------------------------------------|--------------------------------------|--|
| <input type="checkbox"/> A a \$y c d | <input type="checkbox"/> B a \$y c d | <input type="checkbox"/> C \$x y c d | <input type="checkbox"/> D \$x \$y \$z |
|--------------------------------------|--------------------------------------|--------------------------------------|--|

**Question 15 ♣** Soit le script shell suivant :

```
#!/bin/sh
```

```
x=9
x="9*$x"
echo "$x"
```

```
y=0
y="$(echo "$y")"
echo "$y"
```

```
z=1
z="$(echo '$z ')"
echo "$z"
```

Quelles lignes apparaissent, entre autres, dans sa sortie ?

- A 1       B \$x       C 81       D \$y       E 0       F \$z       G 9\*9

**Question 16 ♣** Le répertoire courant contient le fichier « Mon rapport.txt ». Vous avez tous les droits d'accès sur le répertoire courant et ses fichiers. Lesquels des énoncés suivants sont vrais ?

- A La commande « a="Mon rapport.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 B La commande « a=Mon rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 C La commande « a=Mon\ rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 D La commande « a="Mon"; rm "\$a"\* » supprime le fichier « Mon rapport.txt »  
 E La commande « a="Mon rapport.txt"; rm \$a » supprime le fichier « Mon rapport.txt »  
 F La commande « a="Mon\*.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »

**Question 17** Le script shell suivant :

```
#!/bin/sh
```

```
cd /bin
pwd > /tmp/pwd.log
( cd /etc; pwd; )
pwd >> /tmp/pwd.log
{ cd /usr; pwd; }
pwd >> /tmp/pwd.log
{ cd /tmp; pwd; } | rev
pwd >> /tmp/pwd.log
```

a la sortie suivante :

```
/etc
/usr
pmt/
```

Quelle sera le contenu du fichier « /tmp/pwd.log » ?

- A /bin       B /bin       C /bin       D /bin       E /bin       F /bin       G /bin  
 /bin       /bin       /etc       /bin       /bin       /bin       /etc  
 /bin       /bin       /usr       /usr       /usr       /bin       /etc  
 /bin       /tmp       /usr       /tmp       /usr       /usr       /tmp

**Question 18 ♣** La commande « `cmd` » présente dans le PATH a toujours le même comportement et produit la sortie suivante dans le terminal :

```
abc
xbz
```

Parmis les énoncés suivants, lesquels sont forcément vrais ?

- A Il est possible que la conduite « `cmd | grep q` » affiche la ligne « `abc` » dans le terminal.
- B Si « `cmd 2> /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.
- C Si « `cmd | grep q` » n’affiche aucune ligne, alors « `cmd > /dev/null` » n’affiche rien dans le terminal.
- D Si « `cmd > /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.

**Question 19 ♣** Vous avez tous les droits d’accès sur le répertoire courant qui est vide. La commande « `oui` » produit toujours le texte « `pomme` » sur sa sortie standard. La commande « `non` » produit toujours le texte « `pomme` » sur sa sortie d’erreur. Suites à quelles commandes le fichier « `fruits` » contiendra-t-il le mot « `pomme` » ?

- |  |   |   |
|--|---|---|
| <input type="checkbox"/> A <code>oui &gt;&amp;2 2&gt; fruits</code>  | <input type="checkbox"/> C <code>non 2&gt;&amp;1 &gt; fruits</code> | <input type="checkbox"/> E <code>oui &gt; fruits &gt;&amp;2</code>  |
| <input type="checkbox"/> B <code>non 2&gt; fruits 1&gt;&amp;2</code> | <input type="checkbox"/> D <code>oui &gt; fruits 2&gt;&amp;1</code> | <input type="checkbox"/> F <code>non &gt; fruits 2&gt;&amp;1</code> |

**Question 20 ♣** Soit les commandes suivantes, où chaque commande est identifiée par un commentaire :

```
sleep 1; sleep 1           # C1
sleep 1 || sleep 2        # C2
sleep 1 | sleep 2         # C3
{ sleep 2; sleep 2; } | sleep 2 # C4
sleep 2 && sleep 2         # C5
sleep 3 & sleep 4 &       # C6
```

Lesquels des énoncés suivants sont vrais ?

- |  |  |
|--|--|
| <input type="checkbox"/> A La commande C4 dure environ 6 secondes. | <input type="checkbox"/> E La commande C6 dure environ 7 secondes. |
| <input type="checkbox"/> B La commande C5 dure environ 4 secondes. | <input type="checkbox"/> F La commande C3 dure environ 2 secondes. |
| <input type="checkbox"/> C La commande C4 dure environ 4 secondes. | <input type="checkbox"/> G La commande C2 dure environ 3 seconde.  |
| <input type="checkbox"/> D La commande C2 dure environ 2 seconde.  | <input type="checkbox"/> H La commande C1 dure environ 2 secondes. |

**Question 21 ♣** La sortie de « `echo "$PATH"` » est « `/usr/bin:/bin` ». La sortie de « `type rm` » est « `rm est /usr/bin/rm` ». « `/usr/bin/rm` » et « `/bin/sh` » sont conforme au standard POSIX. Vous avez tous les droits d’accès sur « `/a` » et « `/a/rm` » et le script « `/a/rm` » a le contenu suivant :

```
#!/bin/sh
echo "rm"
```

Lesquels des énoncés suivants sont vrais ?

- A La commande « `/a/rm rm` » affiche exactement le texte « `rm` ».
- B La commande « `PATH="/a/rm" rm rm` » affiche exactement le texte « `rm` ».
- C La commande « `rm rm` » affiche exactement le texte « `rm` ».
- D La commande « `cd /a; rm rm` » affiche exactement le texte « `rm` ».
- E La commande « `PATH="/a:/usr/bin:/bin" rm rm` » affiche exactement le texte « `rm` ».
- F La commande « `/bin/sh /a/rm` » affiche exactement le texte « `rm` ».
- G La commande « `export PATH="/a"; rm` » affiche exactement le texte « `rm` ».

**Question 22 ♣** Parmi les commandes suivantes, lesquelles produisent systématiquement la sortie suivante ?

```
abc
def
```

- |  |   |
|--|---|
| <input type="checkbox"/> A ( echo def; echo abc; )   tac | <input type="checkbox"/> D { echo def; echo abc; } > tac  |
| <input type="checkbox"/> B echo def   echo abc   tac     | <input type="checkbox"/> E echo abc   tac; echo def   tac |
| <input type="checkbox"/> C { echo def; echo abc; }   tac | <input type="checkbox"/> F echo def; echo abc;   tac      |

**Question 23 ♣** Vous avez tous les droits d'accès sur le répertoire « /final » et son fichier « /final/banane ». Lesquels des énoncés suivants sont vrais ?

- |   |   |
|---|---|
| <input type="checkbox"/> A « ( exit 1 ) && echo "banane" » affiche le texte « banane ».                 | <input type="checkbox"/> D « rm /final/banane && echo "banane" » supprime le fichier « /final/banane ». |
| <input type="checkbox"/> B « rm /final/banane    echo "banane" » affiche le texte « banane ».           | <input type="checkbox"/> E « rm /final/banane && echo "banane" » affiche le texte « banane ».           |
| <input type="checkbox"/> C « rm /final/banane    echo "banane" » supprime le fichier « /final/banane ». | <input type="checkbox"/> F « ( exit 0 )    echo "banane" » affiche le texte « banane ».                 |

## 5 Scriptage

**Question 24** Soit la mise en situation suivante. Quelle sera la sortie produite par « ./test.sh » à la fin ?

```
rak@koios:~$ ls -l
total 5
-rwxrwxr-x 1 rak rak 134 2025-04-13 20:18 test.sh
rak@koios:~$ cat test.sh
#!/bin/sh

if echo "test" | grep t* > /dev/null; then
    echo "1"
else
    echo "2"
fi

if echo "abcd" | grep t* > /dev/null; then
    echo "3"
else
    echo "4"
fi
rak@koios:~$ ./test.sh
```

- |                                   |                                   |                                   |                                   |
|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| <input type="checkbox"/> A 2<br>4 | <input type="checkbox"/> B 2<br>3 | <input type="checkbox"/> C 1<br>4 | <input type="checkbox"/> D 1<br>3 |
|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|

**Question 25** Soit le script « ./mystere » suivant :

```
#!/bin/bash
x=0
y=3
f() {
    local x=$((x + 1))
    y=$((y - 1))
}
for a in "$@"; do f; done
echo "x:$x y:$y"
```

Quelle sera la sortie de « ./mystere a b c » ?

A x:3 y:0

B x:3 y:3

C x:0 y:0

D x:0 y:3

**Question 26** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
c=""
for f in pomme banane ananas; do
    if [ "$f" = "ananas" ]; then
        exit 1
    fi
    c="$f $c"
done
echo "$c"
```

A Aucune sortie

B ananas banane pomme

C 1

D pomme banane ananas

**Question 27 ♣** Vous avez toutes les permissions sur le répertoire courant. Le script « ./map » a les permissions 555 et le contenu suivant :

```
#!/bin/sh
F="$1"
shift
for x in "$@"; do
    $F "$x"
done
```

Parmi les énoncés suivants, lesquels sont vrais ?

A « ./map echo "Bonjour INF1070" » produit la ligne « Bonjour INF1070 ».

B « ./map echo "Bonjour INF1070" » produit une ligne, avec seulement « Bonjour ».

C « ./map echo a b c » produit, entre autres, la ligne « a ».

D « ./map head \* » produit, entre autres, la ligne « #!/bin/sh ».

E « ./map "echo Bonjour INF1070" » produit, entre autres, la ligne « Bonjour ».

**Question 28** Soient le script shell « `./script1` » suivant :

```
#!/bin/sh
exit 123
```

et le script shell « `./script2` » suivant :

```
#!/bin/sh
if ./script1; then
    echo "$?"
else
    echo "$?"
    echo "$?"
fi
```

Les deux fichiers sont exécutables. Quelle sera la sortie de « `./script2` » ?

A 123  
 B 0

C Aucune sortie

D 123  
0

E 123  
123

## 6 Réseau

**Question 29** Quel protocole se charge de router des paquets de données entre interfaces réseaux de machines connectées à internet ?

A Le protocole ADDR  
 B Le protocole IP  
 C Le protocole SSH

D Le protocole DNS  
 E Le protocole DHCP  
 F Le protocole HTTP

**Question 30** Si je visite le site `https://inf1070.uqam.ca`, Firefox m'affiche un cadenas à côté de l'URL dans le navigateur. Cela signifie que :

A La page n'est pas modifiable par l'utilisateur  
 B La page ne sera jamais modifiée  
 C La connexion s'est faite au travers du protocole DNS

D La connexion est sécurisée au travers d'un certificat  
 E Il faut entrer un mot de passe pour accéder au contenu de la page

## Annexe détachable

**bg** [*job\_id*] Exécute une tâche en arrière plan

**cat** Concaténer des fichiers et les afficher sur la sortie standard. Si aucun fichier n'est fourni, ou si le fichier est « - », **cat** lit l'entrée standard.

**cd** [*rep*] Change le répertoire de travail en cours par *rep*.

### Descripteurs de fichiers

1 = sortie standard, 2 = sortie standard d'erreur

**echo** Affiche une ligne de texte.

**fg** [*job\_id*] Exécute une tâche en avant plan

**exit** Termine le shell avec un code de retour. Si omis, le code de retour est celui de la dernière commande exécutée.

**for** NOM in MOTS ...; do COMMANDES; done

Exécute des commandes pour chaque membre d'une liste. Pour chaque élément dans MOTS, NOM est défini à cet élément, et les COMMANDES sont exécutées. Le code de sortie est celui la dernière commande exécutée.

**grep** Recherche les lignes correspondant à un motif donné. Par défaut, **grep** affiche les lignes qui contiennent une correspondance au motif. Code de retour : 0 si une ligne a été trouvée, 1 sinon.

**head** Afficher le début du contenu de fichiers.

**if** Exécute des commandes selon une condition.

Usage : **if** *lst1*; **then** *lst2*; [ **else** *lst3*; ] **fi**

La liste « *lst1* » est exécutée. Si elle se termine avec le code zéro, alors la liste « *lst2* » est exécutée. Sinon, la liste « *lst3* » est exécutée si elle existe. Le code de retour de l'ensemble est celui de la dernière commande exécutée ou zéro si aucune condition n'était vraie.

**jobs** Affiche les tâches qui sont exécutées.

-1 : donne plus d'information sur les tâches, incluant le numéro de la tâche et son PID

**kill** Envoie un signal à des processus. Les processus sont

identifié par leurs PID ou par des spécifications de tâches (préfixées par %). Si on ne précise pas de signal, TERM est envoyé, qui demande au processus de se terminer.

**ls** Afficher le contenu de répertoires.

-1 : utiliser un format d'affichage long.

**PATH** Variable utilisée par le shell

Elle contient la liste ordonnée des répertoires où le shell va chercher ses commandes avoir vérifié ses primitives.

### Paramètres

**\$#** : le nombre de paramètres positionnels.

**\$@** : l'ensemble des paramètres positionnels sous forme de liste.

**ps** Affiche un instantané des processus en cours. Options :

-e : affiche tous les processus

-f : liste plus de colonnes

-j : liste encore plus de colonnes

**pwd** Afficher le nom du répertoire de travail actuel.

**rm** Effacer des fichiers et des répertoires. Code de retour : 0 en cas de succès, 1 en cas d'échec.

**sed** Éditeur de flux pour filtrer et transformer du texte.

-E utilise des expressions régulières étendues.

**shift** Décale la liste des arguments d'une position vers la gauche.

**sleep** Attendre une durée déterminée (en secondes).

**tac** Concatène et affiche des fichiers à l'envers (la dernière ligne en premier).

**test**, [

Teste l'expression entre crochets [ ] et renvoie un état de sortie correspondant au résultat de la comparaison (0 pour vrai, 1 pour faux).

**arg1 = arg2** : Teste l'égalité entre deux chaînes de caractères.





Remplissez complètement la case (■) pour indiquer une bonne réponse (pas de ×, pas de ✓, pas de cercles).  
Les réponses aux questions sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte.

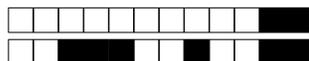
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

← Codez les 8 chiffres de votre code permanent ci-contre, et inscrivez-le à nouveau ci-dessous avec vos nom et prénom. **Les copies sans code permanent ne seront pas corrigées.**

Code permanent :
.....
Prénom :
.....
Nom :
.....

- Question 1 :  A  B  C  D  E  F
- Question 2 :  A  B  C  D  E  F  G  H
- Question 3 :  A  B  C  D
- Question 4 :  A  B  C  D  E  F  G  H
- Question 5 :  A  B  C  D  E  F  G  H
- Question 6 :  A  B  C  D  E
- Question 7 :  A  B  C  D  E
- Question 8 :  A  B  C  D  E  F
- Question 9 :  A  B  C  D  E  F
- Question 10 :  A  B  C  D
- Question 11 :  A  B  C  D  E
- Question 12 :  A  B  C  D  E  F  G  H  I
- Question 13 :  A  B  C  D  E  F  G  H  I
- Question 14 :  A  B  C  D
- Question 15 :  A  B  C  D  E  F  G

- Question 16 :  A  B  C  D  E  F
- Question 17 :  A  B  C  D  E  F  G
- Question 18 :  A  B  C  D
- Question 19 :  A  B  C  D  E  F
- Question 20 :  A  B  C  D  E  F  G  H
- Question 21 :  A  B  C  D  E  F  G
- Question 22 :  A  B  C  D  E  F
- Question 23 :  A  B  C  D  E  F
- Question 24 :  A  B  C  D
- Question 25 :  A  B  C  D
- Question 26 :  A  B  C  D
- Question 27 :  A  B  C  D  E
- Question 28 :  A  B  C  D  E
- Question 29 :  A  B  C  D  E  F
- Question 30 :  A  B  C  D  E



+3/14/19+

INF1070 Utilisation et administration des systèmes informatiques  
Examen Final — Hiver 2025  
Samedi 26 avril — Durée 3 heures  
Ryan Kavanagh et Quentin Stiévenart

- Répondez sur le formulaire (dernière page). Seulement la dernière page sera prise en compte pour la correction.
- Notez aussi votre nom ci-dessous.
- Aucun document ni appareil électronique n'est autorisé.
- L'annexe est détachable.
- Les questions faisant apparaître le symbole ♣ peuvent avoir plusieurs bonnes réponses. Les autres ont une unique bonne réponse.
- Chacune des questions recevra un score entre 0 et 5 points et toutes les questions ont la même pondération.
- Les mauvaises réponses sont pénalisées. Le score d'une question ne peut pas devenir négatif avec les pénalités.
- Le formulaire (dernière page) est lié à votre copie. L'ordre des réponses est différent d'une copie à l'autre. Il ne faut donc pas mélanger le formulaire de réponse de deux copies différentes.

Code permanent :

.....

Prénom :

.....

Nom :

.....

# 1 Expressions régulières

**Question 1 ♣** Les codes postaux canadiens sont des codes alphanumériques formés de six caractères avec le format « LNL NLN », où L est une lettre entre A et Z et N est un chiffre entre 0 et 9. Il y a un espace entre les deux groupes (« LNL », espace, « NLN »). Par exemple, « HOH OH0 » et « P5Q 5P7 » sont des codes postaux, mais « A3B4C6 », « AA5 33B » et « Z5 53H » ne le sont pas. Lesquelles des ERE suivantes ont **une correspondance** avec **au moins** un code postal valide ?

- |   |   |
|---|---|
| <input type="checkbox"/> A (.) (.) (.) \1\2\3                   | <input type="checkbox"/> D $\wedge[A-Z][0-9][A-Z][0-9][A-Z][0-9]\$$ |
| <input type="checkbox"/> B (. [0-9] .\\$ ^ . [A-Z] .)           | <input type="checkbox"/> E ([A-Z] ?[0-9]){3}                        |
| <input type="checkbox"/> C $[\wedge0-9][\wedge A-Z][\wedge0-9]$ | <input type="checkbox"/> F ([A-Z][0-9] ?){2,}                       |

**Question 2 ♣** Une expression régulière  $R_1$  est **équivalente** à une expression régulière  $R_2$  si, pour toute chaîne de caractères  $C$ ,  $R_1$  a une correspondance avec  $C$  si et seulement si  $R_2$  a une correspondance avec  $C$ . Concrètement, cela veut dire que, pour toute entrée, `grep 'R1'` et `grep 'R2'` sélectionneront les mêmes lignes. Indiquez toutes les paires d'expressions régulières étendues (ERE) équivalentes :

- |  |   |
|--|---|
| <input type="checkbox"/> A (a?)+ et a*                   | <input type="checkbox"/> E (a+)? et a*                            |
| <input type="checkbox"/> B $\wedge . * \$$ et $.*$       | <input type="checkbox"/> F a(b c)d et (abd acd)                   |
| <input type="checkbox"/> C a?(b c)* et (a?b a?c)*        | <input type="checkbox"/> G ([ab])[ab] et ([ab])\1                 |
| <input type="checkbox"/> D a(bc d)   (bc d) et a(bc bc)d | <input type="checkbox"/> H $[\wedge abc \$]$ et $[\wedge abc] \$$ |

**Question 3 ♣** Chaque ligne du fichier courriels.html est un lien vers une URL avec le schéma mailto, comme suit :

```
<a href="mailto:alphonse@desjardins.com">Alphonse Desjardins</a>
<a href="mailto:joseph-armand@bombardier.com">Joseph-Armand "Jo" Bombardier</a>
<a href="mailto:mary@la-bolduc.ca" rel="external">Mary Travers</a>
<a href="mailto:paul@manouvellemaison.ca">Paul de Chomedey de Maisonneuve</a>
```

Parmi les commandes suivantes, lesquelles permettent d'extraire de ce fichier toutes les adresses courriels et seulement celles-ci, sans les guillemets ?

- |  |
|--|
| <input type="checkbox"/> A sed -E 's/.*mailto:(["]+).*/\1/' courriels.html           |
| <input type="checkbox"/> B sed -E 's/".*// ' courriels.html   sed -E 's/.*://'       |
| <input type="checkbox"/> C sed -E 's/.*mailto:// ' courriels.html   sed -E 's/".*//' |
| <input type="checkbox"/> D sed -E 's/.*mailto:(.*)".*/\1/' courriels.html            |

**Question 4 ♣** Lesquelles des URLs suivantes ont une correspondance avec l'ERE « ((f|ht)tps?://|mailto:)(infor?|ma|thé)+(tiques?)?.uqam.ca » ?

- |  |   |
|--|---|
| <input type="checkbox"/> A mailto:thétiques?.uqam.ca | <input type="checkbox"/> E mailto:thé@uqam.ca           |
| <input type="checkbox"/> B ftp://info.uqam.ca        | <input type="checkbox"/> F ://ma.uqam.ca                |
| <input type="checkbox"/> C https://uqam.ca           | <input type="checkbox"/> G http://mathématiques.uqam.ca |
| <input type="checkbox"/> D HTTP://MA.UQAM.CA         | <input type="checkbox"/> H https://informatique.uqam.ca |

**Question 5 ♣** Lesquelles des ERE suivantes ont une correspondance avec la chaîne « épaisissait » ?

- |  |  |
|--|--|
| <input type="checkbox"/> A $\wedge (.) . * \wedge 1$   | <input type="checkbox"/> E $(.) \wedge 1 \wedge 1$       |
| <input type="checkbox"/> B (is) . + \wedge 1           | <input type="checkbox"/> F $((.) .) \wedge 1$            |
| <input type="checkbox"/> C $(.) (.) \wedge 2 \wedge 1$ | <input type="checkbox"/> G $(.) . * \wedge 1$            |
| <input type="checkbox"/> D $((.) .) \wedge 2$          | <input type="checkbox"/> H $(.) (.) . \wedge 2 \wedge 1$ |

## 2 Administration

**Question 6 ♣** Parmi les éléments suivants, lesquels font partie des tâches d'un gestionnaire de paquets ?

- A Surveiller en temps réel l'utilisation CPU et RAM des logiciels installés
- B Désinstaller un paquet
- C Installer des fichiers exécutables
- D Installer les dépendances d'un paquet
- E Compter les paquets réseau

**Question 7 ♣** Parmi les énoncés suivants, lesquels sont forcément vrais pour un démon (daemon) ?

- A Il s'exécute en avant plan et interagit directement avec l'utilisateur
- B Il est automatiquement fermé dès que l'utilisateur ferme son terminal
- C Il s'exécute en arrière plan
- D Il s'exécute avec une priorité élevée
- E Il peut être lancé automatiquement au démarrage du système

**Question 8 ♣** Parmi les énoncés suivants, lesquels sont vrais ?

- A Les fichiers de journalisation contiennent des nouvelles importantes sur les nouveaux paquets logiciels disponibles
- B Les journaux des services sont généralement stockés dans `/etc/log/`
- C Certains fichiers de journalisation peuvent être automatiquement archivés
- D Les fichiers de journalisation peuvent être généralement analysés avec des outils comme `grep` pour rechercher des événements spécifiques
- E Les journaux des services sont généralement stockés dans le dossier utilisateur
- F Les journaux des services sont généralement stockés dans `/var/log/`

### 3 Processus

Les 4 questions suivantes se font dans le contexte suivant.

```
$ ps -efj
UID      PID  PPID  PGID   SID  C  STIME TTY      TIME CMD
root      1    0     1     1  0  11:33 ?        00:00:00 /sbin/init
root     507    1   507   507  0  11:33 ?        00:00:00 /usr/sbin/anacron -d -q -s
avahi    508    1   508   508  0  11:33 ?        00:00:00 avahi-daemon: running [debian.local]
root     509    1   509   509  0  11:33 ?        00:00:00 /usr/sbin/cron -f
message+ 510    1   510   510  0  11:33 ?        00:00:00 /usr/bin/dbus-daemon --system --address=sys
root     512    1   512   512  0  11:33 ?        00:00:00 /usr/libexec/low-memory-monitor
debian  1845  1768  1845  1768  0  11:34 pts/0    00:00:00 tmux
debian  1847    1  1847  1847  0  11:34 ?        00:00:00 tmux
debian  1848  1847  1848  1848  0  11:34 pts/1    00:00:00 -bash
debian  1858  1848  1858  1848  0  11:35 pts/1    00:00:00 sleep 1200
debian  1859  1847  1859  1859  0  11:35 pts/2    00:00:00 -bash
debian  1865  1859  1865  1859  1  11:35 pts/2    00:00:06 htop
debian  1866  1847  1866  1866  0  11:35 pts/3    00:00:00 -bash
debian  1951  1847  1951  1951  0  11:41 pts/4    00:00:00 -bash
debian  1957  1951  1957  1951  0  11:41 pts/4    00:00:00 vim
debian  1959  1957  1957  1951  0  11:41 pts/4    00:00:00 cat
debian  1960  1866  1960  1866  0  11:41 pts/3    00:00:00 ps -efj
```

**Question 9 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |   |  |
|---|--|
| <input type="checkbox"/> A L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>sleep</code>      | <input type="checkbox"/> D L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code>   |
| <input type="checkbox"/> B L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code> | <input type="checkbox"/> E L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>sleep</code>        |
| <input type="checkbox"/> C L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>sleep</code>       | <input type="checkbox"/> F L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>avahi-daemon</code> |

**Question 10** Parmi les affirmations suivantes, laquelle est vraie pour le contexte donné ?

- |  |   |
|--|---|
| <input type="checkbox"/> A L'identifiant de processus de la commande <code>sleep</code> est 1858 | <input type="checkbox"/> C L'identifiant de processus de la commande <code>sleep</code> est 11:35 |
| <input type="checkbox"/> B L'identifiant de processus de la commande <code>sleep</code> est 1848 | <input type="checkbox"/> D L'identifiant de processus de la commande <code>sleep</code> est pts/1 |

**Question 11 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |   |   |
|---|---|
| <input type="checkbox"/> A La commande <code>cat</code> a été lancée directement depuis <code>vim</code>  | <input type="checkbox"/> D la commande <code>ps -efj</code> a été lancée directement depuis <code>bash</code> |
| <input type="checkbox"/> B la commande <code>cat</code> a été lancée directement depuis <code>bash</code> | <input type="checkbox"/> E la commande <code>ps -efj</code> a été lancée directement depuis <code>vim</code>  |
| <input type="checkbox"/> C la commande <code>cat</code> a été lancée directement depuis <code>tmux</code> |   |

Pour les questions suivantes, on a le contexte suivant.

```
$ jobs -l
[1] 3532  Running gnome-calculator &
[2] 3643  Running nautilus &
[3] 3710  Running xeyes &
```

**Question 12 ♣** Après avoir effectué avec succès la commande `kill 3643`, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>gnome-calculator</code> s'exécute en avant-plan   | <input type="checkbox"/> E Le processus <code>nautilus</code> s'exécute en avant-plan |
| <input type="checkbox"/> B Le processus <code>nautilus</code> s'exécute en arrière-plan         | <input type="checkbox"/> F Le processus <code>xeyes</code> s'exécute en arrière-plan  |
| <input type="checkbox"/> C Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> G Le processus <code>nautilus</code> est terminé             |
| <input type="checkbox"/> D Le processus <code>xeyes</code> s'exécute en avant-plan              | <input type="checkbox"/> H Le processus <code>gnome-calculator</code> est terminé     |
|   | <input type="checkbox"/> I Le processus <code>xeyes</code> est terminé                |

**Question 13 ♣** À la place de la commande de la question précédente, si j'avais effectué la commande `fg %3` et ensuite entré `^C` (Contrôle-C) au clavier, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> F Le processus <code>nautilus</code> s'exécute en arrière-plan       |
| <input type="checkbox"/> B Le processus <code>xeyes</code> est terminé                          | <input type="checkbox"/> G Le processus <code>xeyes</code> s'exécute en arrière-plan          |
| <input type="checkbox"/> C Le processus <code>nautilus</code> s'exécute en avant-plan           | <input type="checkbox"/> H Le processus <code>gnome-calculator</code> s'exécute en avant-plan |
| <input type="checkbox"/> D Le processus <code>nautilus</code> est terminé                       | <input type="checkbox"/> I Le processus <code>xeyes</code> s'exécute en avant-plan            |
| <input type="checkbox"/> E Le processus <code>gnome-calculator</code> est terminé               |   |

## 4 Shell avancé

**Question 14** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
x="a"
y="b"
z="c      d"
echo "$x" '$y' $z
```

- |                                      |  |   |                                      |
|--------------------------------------|--|---|--------------------------------------|
| <input type="checkbox"/> A a \$y c d | <input type="checkbox"/> B \$x \$y \$z | <input type="checkbox"/> C a \$y c      d | <input type="checkbox"/> D \$x y c d |
|--------------------------------------|--|---|--------------------------------------|

**Question 15 ♣** Soit le script shell suivant :

```
#!/bin/sh
```

```
x=9
x="9*$x"
echo "$x"
```

```
y=0
y="$(echo "$y")"
echo "$y"
```

```
z=1
z="$(echo '$z ')"
echo "$z"
```

Quelles lignes apparaissent, entre autres, dans sa sortie ?

- A \$x       B 0       C 1       D \$z       E 9\*9       F 81       G \$y

**Question 16 ♣** Le répertoire courant contient le fichier « Mon rapport.txt ». Vous avez tous les droits d'accès sur le répertoire courant et ses fichiers. Lesquels des énoncés suivants sont vrais ?

- A La commande « a=Mon\ rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 B La commande « a="Mon rapport.txt"; rm \$a » supprime le fichier « Mon rapport.txt »  
 C La commande « a="Mon"; rm "\$a"\* » supprime le fichier « Mon rapport.txt »  
 D La commande « a=Mon rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 E La commande « a="Mon rapport.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 F La commande « a="Mon\*.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »

**Question 17** Le script shell suivant :

```
#!/bin/sh
```

```
cd /bin
pwd > /tmp/pwd.log
( cd /etc; pwd; )
pwd >> /tmp/pwd.log
{ cd /usr; pwd; }
pwd >> /tmp/pwd.log
{ cd /tmp; pwd; } | rev
pwd >> /tmp/pwd.log
```

a la sortie suivante :

```
/etc
/usr
pmt/
```

Quelle sera le contenu du fichier « /tmp/pwd.log » ?

- A /bin       B /bin       C /bin       D /bin       E /bin       F /bin       G /bin  
           /etc            /bin            /bin            /bin            /bin            /bin            /etc  
           /etc            /usr            /bin            /bin            /bin            /usr            /usr  
           /tmp            /tmp            /tmp            /bin            /usr            /usr            /usr

**Question 18 ♣** La commande « `cmd` » présente dans le `PATH` a toujours le même comportement et produit la sortie suivante dans le terminal :

```
abc
xbz
```

Parmi les énoncés suivants, lesquels sont forcément vrais ?

- A Il est possible que la conduite « `cmd | grep q` » affiche la ligne « `abc` » dans le terminal.
- B Si « `cmd > /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.
- C Si « `cmd | grep q` » n’affiche aucune ligne, alors « `cmd > /dev/null` » n’affiche rien dans le terminal.
- D Si « `cmd 2> /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.

**Question 19 ♣** Vous avez tous les droits d’accès sur le répertoire courant qui est vide. La commande « `oui` » produit toujours le texte « `pomme` » sur sa sortie standard. La commande « `non` » produit toujours le texte « `pomme` » sur sa sortie d’erreur. Suites à quelles commandes le fichier « `fruits` » contiendra-t-il le mot « `pomme` » ?

- |  |   |   |
|--|---|---|
| <input type="checkbox"/> A <code>non 2&gt; fruits 1&gt;&amp;2</code> | <input type="checkbox"/> C <code>non &gt; fruits 2&gt;&amp;1</code> | <input type="checkbox"/> E <code>oui &gt; fruits &gt;&amp;2</code>  |
| <input type="checkbox"/> B <code>oui &gt;&amp;2 2&gt; fruits</code>  | <input type="checkbox"/> D <code>non 2&gt;&amp;1 &gt; fruits</code> | <input type="checkbox"/> F <code>oui &gt; fruits 2&gt;&amp;1</code> |

**Question 20 ♣** Soit les commandes suivantes, où chaque commande est identifiée par un commentaire :

```
sleep 1; sleep 1           # C1
sleep 1 || sleep 2        # C2
sleep 1 | sleep 2         # C3
{ sleep 2; sleep 2; } | sleep 2 # C4
sleep 2 && sleep 2         # C5
sleep 3 & sleep 4 &       # C6
```

Lesquels des énoncés suivants sont vrais ?

- |  |  |
|--|--|
| <input type="checkbox"/> A La commande C4 dure environ 4 secondes. | <input type="checkbox"/> E La commande C1 dure environ 2 secondes. |
| <input type="checkbox"/> B La commande C3 dure environ 2 secondes. | <input type="checkbox"/> F La commande C4 dure environ 6 secondes. |
| <input type="checkbox"/> C La commande C5 dure environ 4 secondes. | <input type="checkbox"/> G La commande C2 dure environ 3 seconde.  |
| <input type="checkbox"/> D La commande C2 dure environ 2 seconde.  | <input type="checkbox"/> H La commande C6 dure environ 7 secondes. |

**Question 21 ♣** La sortie de « `echo "$PATH"` » est « `/usr/bin:/bin` ». La sortie de « `type rm` » est « `rm est /usr/bin/rm` ». « `/usr/bin/rm` » et « `/bin/sh` » sont conforme au standard POSIX. Vous avez tous les droits d’accès sur « `/a` » et « `/a/rm` » et le script « `/a/rm` » a le contenu suivant :

```
#!/bin/sh
echo "rm"
```

Lesquels des énoncés suivants sont vrais ?

- A La commande « `cd /a; rm rm` » affiche exactement le texte « `rm` ».
- B La commande « `/a/rm rm` » affiche exactement le texte « `rm` ».
- C La commande « `PATH="/a/rm" rm rm` » affiche exactement le texte « `rm` ».
- D La commande « `rm rm` » affiche exactement le texte « `rm` ».
- E La commande « `PATH="/a:/usr/bin:/bin" rm rm` » affiche exactement le texte « `rm` ».
- F La commande « `export PATH="/a"; rm` » affiche exactement le texte « `rm` ».
- G La commande « `/bin/sh /a/rm` » affiche exactement le texte « `rm` ».

**Question 22 ♣** Parmi les commandes suivantes, lesquelles produisent systématiquement la sortie suivante ?

```
abc
def
```

- |                            |                               |                            |                                |
|----------------------------|-------------------------------|----------------------------|--------------------------------|
| <input type="checkbox"/> A | echo def   echo abc   tac     | <input type="checkbox"/> D | echo def; echo abc;   tac      |
| <input type="checkbox"/> B | ( echo def; echo abc; )   tac | <input type="checkbox"/> E | echo abc   tac; echo def   tac |
| <input type="checkbox"/> C | { echo def; echo abc; }   tac | <input type="checkbox"/> F | { echo def; echo abc; } > tac  |

**Question 23 ♣** Vous avez tous les droits d'accès sur le répertoire « /final » et son fichier « /final/banane ». Lesquels des énoncés suivants sont vrais ?

- |                            |  |                            |  |
|----------------------------|--|----------------------------|--|
| <input type="checkbox"/> A | « ( exit 1 ) && echo "banane" » affiche le texte « banane ».                 | <input type="checkbox"/> D | « rm /final/banane && echo "banane" » affiche le texte « banane ». |
| <input type="checkbox"/> B | « rm /final/banane && echo "banane" » supprime le fichier « /final/banane ». | <input type="checkbox"/> E | « ( exit 0 )    echo "banane" » affiche le texte « banane ».       |
| <input type="checkbox"/> C | « rm /final/banane    echo "banane" » supprime le fichier « /final/banane ». | <input type="checkbox"/> F | « rm /final/banane    echo "banane" » affiche le texte « banane ». |

## 5 Scriptage

**Question 24** Soit la mise en situation suivante. Quelle sera la sortie produite par « ./test.sh » à la fin ?

```
rak@koios:~$ ls -l
total 5
-rwxrwxr-x 1 rak rak 134 2025-04-13 20:18 test.sh
rak@koios:~$ cat test.sh
#!/bin/sh

if echo "test" | grep t* > /dev/null; then
    echo "1"
else
    echo "2"
fi

if echo "abcd" | grep t* > /dev/null; then
    echo "3"
else
    echo "4"
fi
rak@koios:~$ ./test.sh
```

- |                            |        |                            |        |                            |        |                            |        |
|----------------------------|--------|----------------------------|--------|----------------------------|--------|----------------------------|--------|
| <input type="checkbox"/> A | 1<br>3 | <input type="checkbox"/> B | 1<br>4 | <input type="checkbox"/> C | 2<br>4 | <input type="checkbox"/> D | 2<br>3 |
|----------------------------|--------|----------------------------|--------|----------------------------|--------|----------------------------|--------|

**Question 25** Soit le script « ./mystere » suivant :

```
#!/bin/bash
x=0
y=3
f() {
    local x=$((x + 1))
    y=$((y - 1))
}
for a in "$@"; do f; done
echo "x:$x y:$y"
```

Quelle sera la sortie de « ./mystere a b c » ?

A x:0 y:3

B x:3 y:3

C x:0 y:0

D x:3 y:0

**Question 26** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
c=""
for f in pomme banane ananas; do
    if [ "$f" = "ananas" ]; then
        exit 1
    fi
    c="$f $c"
done
echo "$c"
```

A ananas banane pomme

B 1

C Aucune sortie

D pomme banane ananas

**Question 27 ♣** Vous avez toutes les permissions sur le répertoire courant. Le script « ./map » a les permissions 555 et le contenu suivant :

```
#!/bin/sh
F="$1"
shift
for x in "$@"; do
    $F "$x"
done
```

Parmi les énoncés suivants, lesquels sont vrais ?

A « ./map head \* » produit, entre autres, la ligne « #!/bin/sh ».

B « ./map echo "Bonjour INF1070" » produit la ligne « Bonjour INF1070 ».

C « ./map "echo Bonjour INF1070" » produit, entre autres, la ligne « Bonjour ».

D « ./map echo a b c » produit, entre autres, la ligne « a ».

E « ./map echo "Bonjour INF1070" » produit une ligne, avec seulement « Bonjour ».

**Question 28** Soient le script shell « `./script1` » suivant :

```
#!/bin/sh
exit 123
```

et le script shell « `./script2` » suivant :

```
#!/bin/sh
if ./script1; then
    echo "$?"
else
    echo "$?"
    echo "$?"
fi
```

Les deux fichiers sont exécutables. Quelle sera la sortie de « `./script2` » ?

- |                                |                                     |                                       |  |
|--------------------------------|-------------------------------------|---------------------------------------|--|
| <input type="checkbox"/> A 123 | <input type="checkbox"/> B 123<br>0 | <input type="checkbox"/> C 123<br>123 | <input type="checkbox"/> D Aucune sortie<br><input type="checkbox"/> E 0 |
|--------------------------------|-------------------------------------|---------------------------------------|--|

## 6 Réseau

**Question 29** Quel protocole se charge de router des paquets de données entre interfaces réseaux de machines connectées à internet ?

- |  |   |
|--|---|
| <input type="checkbox"/> A Le protocole DHCP | <input type="checkbox"/> D Le protocole DNS |
| <input type="checkbox"/> B Le protocole HTTP | <input type="checkbox"/> E Le protocole SSH |
| <input type="checkbox"/> C Le protocole ADDR | <input type="checkbox"/> F Le protocole IP  |

**Question 30** Si je visite le site `https://inf1070.uqam.ca`, Firefox m'affiche un cadenas à côté de l'URL dans le navigateur. Cela signifie que :

- |  |  |
|--|--|
| <input type="checkbox"/> A La page n'est pas modifiable par l'utilisateur                    | <input type="checkbox"/> C La connexion est sécurisée au travers d'un certificat |
| <input type="checkbox"/> B Il faut entrer un mot de passe pour accéder au contenu de la page | <input type="checkbox"/> D La page ne sera jamais modifiée                       |
|  | <input type="checkbox"/> E La connexion s'est faite au travers du protocole DNS  |

## Annexe détachable

**bg** [*job\_id*] Exécute une tâche en arrière plan

**cat** Concaténer des fichiers et les afficher sur la sortie standard. Si aucun fichier n'est fourni, ou si le fichier est « - », **cat** lit l'entrée standard.

**cd** [*rep*] Change le répertoire de travail en cours par *rep*.

### Descripteurs de fichiers

1 = sortie standard, 2 = sortie standard d'erreur

**echo** Affiche une ligne de texte.

**fg** [*job\_id*] Exécute une tâche en avant plan

**exit** Termine le shell avec un code de retour. Si omis, le code de retour est celui de la dernière commande exécutée.

**for** NOM in MOTS ...; do COMMANDES; done

Exécute des commandes pour chaque membre d'une liste. Pour chaque élément dans MOTS, NOM est défini à cet élément, et les COMMANDES sont exécutées. Le code de sortie est celui la dernière commande exécutée.

**grep** Recherche les lignes correspondant à un motif donné. Par défaut, **grep** affiche les lignes qui contiennent une correspondance au motif. Code de retour : 0 si une ligne a été trouvée, 1 sinon.

**head** Afficher le début du contenu de fichiers.

**if** Exécute des commandes selon une condition.

Usage : **if** *lst1*; **then** *lst2*; [ **else** *lst3*; ] **fi**

La liste « *lst1* » est exécutée. Si elle se termine avec le code zéro, alors la liste « *lst2* » est exécutée. Sinon, la liste « *lst3* » est exécutée si elle existe. Le code de retour de l'ensemble est celui de la dernière commande exécutée ou zéro si aucune condition n'était vraie.

**jobs** Affiche les tâches qui sont exécutées.

-1 : donne plus d'information sur les tâches, incluant le numéro de la tâche et son PID

**kill** Envoie un signal à des processus. Les processus sont

identifié par leurs PID ou par des spécifications de tâches (préfixées par %). Si on ne précise pas de signal, TERM est envoyé, qui demande au processus de se terminer.

**ls** Afficher le contenu de répertoires.

-l : utiliser un format d'affichage long.

**PATH** Variable utilisée par le shell

Elle contient la liste ordonnée des répertoires où le shell va chercher ses commandes avoir vérifié ses primitives.

### Paramètres

**\$#** : le nombre de paramètres positionnels.

**\$@** : l'ensemble des paramètres positionnels sous forme de liste.

**ps** Affiche un instantané des processus en cours. Options :

-e : affiche tous les processus

-f : liste plus de colonnes

-j : liste encore plus de colonnes

**pwd** Afficher le nom du répertoire de travail actuel.

**rm** Effacer des fichiers et des répertoires. Code de retour : 0 en cas de succès, 1 en cas d'échec.

**sed** Éditeur de flux pour filtrer et transformer du texte.

-E utilise des expressions régulières étendues.

**shift** Décale la liste des arguments d'une position vers la gauche.

**sleep** Attendre une durée déterminée (en secondes).

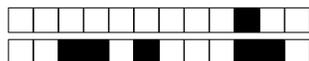
**tac** Concatène et affiche des fichiers à l'envers (la dernière ligne en premier).

**test**, [

Teste l'expression entre crochets [ ] et renvoie un état de sortie correspondant au résultat de la comparaison (0 pour vrai, 1 pour faux).

**arg1 = arg2** : Teste l'égalité entre deux chaînes de caractères.





Remplissez complètement la case (■) pour indiquer une bonne réponse (pas de ×, pas de ✓, pas de cercles). Les réponses aux questions sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte.

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

← Codez les 8 chiffres de votre code permanent ci-contre, et inscrivez-le à nouveau ci-dessous avec vos nom et prénom. Les copies sans code permanent ne seront pas corrigées.

Code permanent :
.....
Prénom :
.....
Nom :
.....

- Question 1 : A B C D E F
- Question 2 : A B C D E F G H
- Question 3 : A B C D
- Question 4 : A B C D E F G H
- Question 5 : A B C D E F G H
- Question 6 : A B C D E
- Question 7 : A B C D E
- Question 8 : A B C D E F
- Question 9 : A B C D E F
- Question 10 : A B C D
- Question 11 : A B C D E
- Question 12 : A B C D E F G H I
- Question 13 : A B C D E F G H I
- Question 14 : A B C D
- Question 15 : A B C D E F G

- Question 16 : A B C D E F
- Question 17 : A B C D E F G
- Question 18 : A B C D
- Question 19 : A B C D E F
- Question 20 : A B C D E F G H
- Question 21 : A B C D E F G
- Question 22 : A B C D E F
- Question 23 : A B C D E F
- Question 24 : A B C D
- Question 25 : A B C D
- Question 26 : A B C D
- Question 27 : A B C D E
- Question 28 : A B C D E
- Question 29 : A B C D E F
- Question 30 : A B C D E



+4/14/5+

INF1070 Utilisation et administration des systèmes informatiques  
Examen Final — Hiver 2025  
Samedi 26 avril — Durée 3 heures  
Ryan Kavanagh et Quentin Stiévenart

- Répondez sur le formulaire (dernière page). Seulement la dernière page sera prise en compte pour la correction.
- Notez aussi votre nom ci-dessous.
- Aucun document ni appareil électronique n'est autorisé.
- L'annexe est détachable.
- Les questions faisant apparaître le symbole ♣ peuvent avoir plusieurs bonnes réponses. Les autres ont une unique bonne réponse.
- Chacune des questions recevra un score entre 0 et 5 points et toutes les questions ont la même pondération.
- Les mauvaises réponses sont pénalisées. Le score d'une question ne peut pas devenir négatif avec les pénalités.
- Le formulaire (dernière page) est lié à votre copie. L'ordre des réponses est différent d'une copie à l'autre. Il ne faut donc pas mélanger le formulaire de réponse de deux copies différentes.

Code permanent :

.....

Prénom :

.....

Nom :

.....

# 1 Expressions régulières

**Question 1 ♣** Les codes postaux canadiens sont des codes alphanumériques formés de six caractères avec le format « LNL NLN », où L est une lettre entre A et Z et N est un chiffre entre 0 et 9. Il y a un espace entre les deux groupes (« LNL », espace, « NLN »). Par exemple, « HOH OH0 » et « P5Q 5P7 » sont des codes postaux, mais « A3B4C6 », « AA5 33B » et « Z5 53H » ne le sont pas. Lesquelles des ERE suivantes ont **une correspondance** avec **au moins** un code postal valide ?

- |   |   |
|---|---|
| <input type="checkbox"/> A $\wedge[A-Z][0-9][A-Z][0-9][A-Z][0-9]\$$ | <input type="checkbox"/> D $(.)(.)(.)\ \backslash1\backslash2\backslash3$ |
| <input type="checkbox"/> B $([A-Z][0-9]?)\{2,\}$                    | <input type="checkbox"/> E $[\wedge0-9][\wedge A-Z][\wedge0-9]$           |
| <input type="checkbox"/> C $(.[0-9].\$ \wedge.[A-Z].)$              | <input type="checkbox"/> F $([A-Z]?[0-9])\{3\}$                           |

**Question 2 ♣** Une expression régulière  $R_1$  est **équivalente** à une expression régulière  $R_2$  si, pour toute chaîne de caractères  $C$ ,  $R_1$  a une correspondance avec  $C$  si et seulement si  $R_2$  a une correspondance avec  $C$ . Concrètement, cela veut dire que, pour toute entrée, `grep 'R1'` et `grep 'R2'` sélectionneront les mêmes lignes. Indiquez toutes les paires d'expressions régulières étendues (ERE) équivalentes :

- |  |   |
|--|---|
| <input type="checkbox"/> A $(a?)^+$ et $a^*$                   | <input type="checkbox"/> E $(a+)?$ et $a^*$                 |
| <input type="checkbox"/> B $([ab])[ab]$ et $([ab])\backslash1$ | <input type="checkbox"/> F $a(b c)d$ et $(abd acd)$         |
| <input type="checkbox"/> C $[\wedge abc\$]$ et $\wedge[abc]\$$ | <input type="checkbox"/> G $\wedge.*\$$ et $.*$             |
| <input type="checkbox"/> D $a?(b c)^*$ et $(a?b a?c)^*$        | <input type="checkbox"/> H $a(b[cd]  [bc]d)$ et $a(bc bc)d$ |

**Question 3 ♣** Chaque ligne du fichier `courriels.html` est un lien vers une URL avec le schéma `mailto`, comme suit :

```
<a href="mailto:alphonse@desjardins.com">Alphonse Desjardins</a>
<a href="mailto:joseph-armand@bombardier.com">Joseph-Armand "Jo" Bombardier</a>
<a href="mailto:mary@la-bolduc.ca" rel="external">Mary Travers</a>
<a href="mailto:paul@manouvellemaison.ca">Paul de Chomedey de Maisonneuve</a>
```

Parmi les commandes suivantes, lesquelles permettent d'extraire de ce fichier toutes les adresses courriels et seulement celles-ci, sans les guillemets ?

- |   |
|---|
| <input type="checkbox"/> A <code>sed -E 's/".*//'</code> <code>courriels.html</code>   <code>sed -E 's/.*:/'</code>       |
| <input type="checkbox"/> B <code>sed -E 's/.*mailto:(.*)".*/\1/'</code> <code>courriels.html</code>                       |
| <input type="checkbox"/> C <code>sed -E 's/.*mailto:/'</code> <code>courriels.html</code>   <code>sed -E 's/".*//'</code> |
| <input type="checkbox"/> D <code>sed -E 's/.*mailto:(["^]+).*/\1/'</code> <code>courriels.html</code>                     |

**Question 4 ♣** Lesquelles des URLs suivantes ont une correspondance avec l'ERE «  $((f|ht)tps?://|mailto:)(infor?|ma|thé)+(tiques?)?.uqam.ca$  » ?

- |  |  |
|--|--|
| <input type="checkbox"/> A <code>https://informatique.uqam.ca</code> | <input type="checkbox"/> E <code>http://mathématiques.uqam.ca</code> |
| <input type="checkbox"/> B <code>https://uqam.ca</code>              | <input type="checkbox"/> F <code>ftp://info.uqam.ca</code>           |
| <input type="checkbox"/> C <code>mailto:thétiques?.uqam.ca</code>    | <input type="checkbox"/> G <code>mailto:thé@uqam.ca</code>           |
| <input type="checkbox"/> D <code>://ma.uqam.ca</code>                | <input type="checkbox"/> H <code>HTTP://MA.UQAM.CA</code>            |

**Question 5 ♣** Lesquelles des ERE suivantes ont une correspondance avec la chaîne « épaisissait » ?

- |  |  |
|--|--|
| <input type="checkbox"/> A $(..)*\backslash1$                | <input type="checkbox"/> E $\wedge(..)*\backslash1$    |
| <input type="checkbox"/> B $((..))\backslash1$               | <input type="checkbox"/> F $(is)^+\backslash1$         |
| <input type="checkbox"/> C $(..)(..)\backslash2\backslash1$  | <input type="checkbox"/> G $((..))\backslash2$         |
| <input type="checkbox"/> D $(..)(..).\backslash2\backslash1$ | <input type="checkbox"/> H $(.)\backslash1\backslash1$ |

## 2 Administration

**Question 6 ♣** Parmi les éléments suivants, lesquels font partie des tâches d'un gestionnaire de paquets ?

- A Installer les dépendances d'un paquet
- B Compter les paquets réseau
- C Désinstaller un paquet
- D Surveiller en temps réel l'utilisation CPU et RAM des logiciels installés
- E Installer des fichiers exécutables

**Question 7 ♣** Parmi les énoncés suivants, lesquels sont forcément vrais pour un démon (daemon) ?

- A Il est automatiquement fermé dès que l'utilisateur ferme son terminal
- B Il s'exécute en avant plan et interagit directement avec l'utilisateur
- C Il peut être lancé automatiquement au démarrage du système
- D Il s'exécute avec une priorité élevée
- E Il s'exécute en arrière plan

**Question 8 ♣** Parmi les énoncés suivants, lesquels sont vrais ?

- A Les journaux des services sont généralement stockés dans `/var/log/`
- B Les journaux des services sont généralement stockés dans `/etc/log/`
- C Certains fichiers de journalisation peuvent être automatiquement archivés
- D Les fichiers de journalisation peuvent être généralement analysés avec des outils comme `grep` pour rechercher des événements spécifiques
- E Les journaux des services sont généralement stockés dans le dossier utilisateur
- F Les fichiers de journalisation contiennent des nouvelles importantes sur les nouveaux paquets logiciels disponibles

### 3 Processus

Les 4 questions suivantes se font dans le contexte suivant.

```
$ ps -efj
UID      PID  PPID  PGID   SID  C  STIME TTY      TIME CMD
root      1    0     1     1  0  11:33 ?        00:00:00 /sbin/init
root     507    1   507   507  0  11:33 ?        00:00:00 /usr/sbin/anacron -d -q -s
avahi    508    1   508   508  0  11:33 ?        00:00:00 avahi-daemon: running [debian.local]
root     509    1   509   509  0  11:33 ?        00:00:00 /usr/sbin/cron -f
message+ 510    1   510   510  0  11:33 ?        00:00:00 /usr/bin/dbus-daemon --system --address=sys
root     512    1   512   512  0  11:33 ?        00:00:00 /usr/libexec/low-memory-monitor
debian  1845  1768  1845  1768  0  11:34 pts/0    00:00:00 tmux
debian  1847    1  1847  1847  0  11:34 ?        00:00:00 tmux
debian  1848  1847  1848  1848  0  11:34 pts/1    00:00:00 -bash
debian  1858  1848  1858  1848  0  11:35 pts/1    00:00:00 sleep 1200
debian  1859  1847  1859  1859  0  11:35 pts/2    00:00:00 -bash
debian  1865  1859  1865  1859  1  11:35 pts/2    00:00:06 htop
debian  1866  1847  1866  1866  0  11:35 pts/3    00:00:00 -bash
debian  1951  1847  1951  1951  0  11:41 pts/4    00:00:00 -bash
debian  1957  1951  1957  1951  0  11:41 pts/4    00:00:00 vim
debian  1959  1957  1957  1951  0  11:41 pts/4    00:00:00 cat
debian  1960  1866  1960  1866  0  11:41 pts/3    00:00:00 ps -efj
```

**Question 9 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |  |
|--|--|
| <input type="checkbox"/> A L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>avahi-daemon</code> | <input type="checkbox"/> D L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code> |
| <input type="checkbox"/> B L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>sleep</code>         | <input type="checkbox"/> E L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>sleep</code>        |
| <input type="checkbox"/> C L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code>    | <input type="checkbox"/> F L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>sleep</code>      |

**Question 10** Parmi les affirmations suivantes, laquelle est vraie pour le contexte donné ?

- |  |  |
|--|--|
| <input type="checkbox"/> A L'identifiant de processus de la commande <code>sleep</code> est <code>pts/1</code> | <input type="checkbox"/> C L'identifiant de processus de la commande <code>sleep</code> est <code>1858</code>  |
| <input type="checkbox"/> B L'identifiant de processus de la commande <code>sleep</code> est <code>1848</code>  | <input type="checkbox"/> D L'identifiant de processus de la commande <code>sleep</code> est <code>11:35</code> |

**Question 11 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |   |
|--|---|
| <input type="checkbox"/> A la commande <code>ps -efj</code> a été lancée directement depuis <code>vim</code> | <input type="checkbox"/> D La commande <code>cat</code> a été lancée directement depuis <code>vim</code>      |
| <input type="checkbox"/> B la commande <code>cat</code> a été lancée directement depuis <code>tmux</code>    | <input type="checkbox"/> E la commande <code>ps -efj</code> a été lancée directement depuis <code>bash</code> |
| <input type="checkbox"/> C la commande <code>cat</code> a été lancée directement depuis <code>bash</code>    |   |

Pour les questions suivantes, on a le contexte suivant.

```
$ jobs -l
[1] 3532 Running gnome-calculator &
[2] 3643 Running nautilus &
[3] 3710 Running xeyes &
```

**Question 12 ♣** Après avoir effectué avec succès la commande `kill 3643`, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>xeyes</code> s'exécute en avant-plan      | <input type="checkbox"/> F Le processus <code>gnome-calculator</code> est terminé               |
| <input type="checkbox"/> B Le processus <code>nautilus</code> s'exécute en avant-plan   | <input type="checkbox"/> G Le processus <code>gnome-calculator</code> s'exécute en avant-plan   |
| <input type="checkbox"/> C Le processus <code>nautilus</code> s'exécute en arrière-plan | <input type="checkbox"/> H Le processus <code>xeyes</code> est terminé                          |
| <input type="checkbox"/> D Le processus <code>xeyes</code> s'exécute en arrière-plan    | <input type="checkbox"/> I Le processus <code>gnome-calculator</code> s'exécute en arrière-plan |
| <input type="checkbox"/> E Le processus <code>nautilus</code> est terminé               |   |

**Question 13 ♣** À la place de la commande de la question précédente, si j'avais effectué la commande `fg %3` et ensuite entré `^C` (Contrôle-C) au clavier, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>xeyes</code> est terminé                          | <input type="checkbox"/> F Le processus <code>xeyes</code> s'exécute en arrière-plan          |
| <input type="checkbox"/> B Le processus <code>nautilus</code> s'exécute en avant-plan           | <input type="checkbox"/> G Le processus <code>nautilus</code> est terminé                     |
| <input type="checkbox"/> C Le processus <code>xeyes</code> s'exécute en avant-plan              | <input type="checkbox"/> H Le processus <code>gnome-calculator</code> s'exécute en avant-plan |
| <input type="checkbox"/> D Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> I Le processus <code>nautilus</code> s'exécute en arrière-plan       |
| <input type="checkbox"/> E Le processus <code>gnome-calculator</code> est terminé               |   |

## 4 Shell avancé

**Question 14** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
x="a"
y="b"
z="c d"
echo "$x" '$y' $z
```

- |                                      |  |                                      |                                      |
|--------------------------------------|--|--------------------------------------|--------------------------------------|
| <input type="checkbox"/> A \$x y c d | <input type="checkbox"/> B \$x \$y \$z | <input type="checkbox"/> C a \$y c d | <input type="checkbox"/> D a \$y c d |
|--------------------------------------|--|--------------------------------------|--------------------------------------|

**Question 15 ♣** Soit le script shell suivant :

```
#!/bin/sh

x=9
x="9*$x"
echo "$x"

y=0
y="$(echo "$y")"
echo "$y"

z=1
z="$(echo '$z ')"
echo "$z"
```

Quelles lignes apparaissent, entre autres, dans sa sortie ?

- A 1       B \$y       C \$z       D \$x       E 0       F 9\*9       G 81

**Question 16 ♣** Le répertoire courant contient le fichier « Mon rapport.txt ». Vous avez tous les droits d'accès sur le répertoire courant et ses fichiers. Lesquels des énoncés suivants sont vrais ?

- A La commande « a="Mon rapport.txt"; rm \$a » supprime le fichier « Mon rapport.txt »  
 B La commande « a="Mon"; rm "\$a"\* » supprime le fichier « Mon rapport.txt »  
 C La commande « a="Mon\*.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 D La commande « a="Mon rapport.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 E La commande « a=Mon rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 F La commande « a=Mon\ rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »

**Question 17** Le script shell suivant :

```
#!/bin/sh

cd /bin
pwd > /tmp/pwd.log
( cd /etc; pwd; )
pwd >> /tmp/pwd.log
{ cd /usr; pwd; }
pwd >> /tmp/pwd.log
{ cd /tmp; pwd; } | rev
pwd >> /tmp/pwd.log
```

a la sortie suivante :

```
/etc
/usr
pmt/
```

Quelle sera le contenu du fichier « /tmp/pwd.log » ?

- A /bin       B /bin       C /bin       D /bin       E /bin       F /bin       G /bin  
           /etc            /bin            /bin            /bin            /bin            /bin            /etc  
           /usr            /bin            /bin            /bin            /usr            /usr            /etc  
           /usr            /usr            /bin            /tmp            /tmp            /usr            /tmp

**Question 18 ♣** La commande « `cmd` » présente dans le `PATH` a toujours le même comportement et produit la sortie suivante dans le terminal :

```
abc
xbz
```

Parmi les énoncés suivants, lesquels sont forcément vrais ?

- A Si « `cmd | grep q` » n'affiche aucune ligne, alors « `cmd > /dev/null` » n'affiche rien dans le terminal.
- B Si « `cmd > /dev/null` » n'affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.
- C Il est possible que la conduite « `cmd | grep q` » affiche la ligne « `abc` » dans le terminal.
- D Si « `cmd 2> /dev/null` » n'affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.

**Question 19 ♣** Vous avez tous les droits d'accès sur le répertoire courant qui est vide. La commande « `oui` » produit toujours le texte « `pomme` » sur sa sortie standard. La commande « `non` » produit toujours le texte « `pomme` » sur sa sortie d'erreur. Suites à quelles commandes le fichier « `fruits` » contiendra-t-il le mot « `pomme` » ?

- A `non > fruits 2>&1`
- B `oui > fruits 2>&1`
- C `oui > fruits >&2`
- D `oui >&2 2> fruits`
- E `non 2>&1 > fruits`
- F `non 2> fruits 1>&2`

**Question 20 ♣** Soit les commandes suivantes, où chaque commande est identifiée par un commentaire :

```
sleep 1; sleep 1           # C1
sleep 1 || sleep 2        # C2
sleep 1 | sleep 2         # C3
{ sleep 2; sleep 2; } | sleep 2 # C4
sleep 2 && sleep 2         # C5
sleep 3 & sleep 4 &       # C6
```

Lesquels des énoncés suivants sont vrais ?

- A La commande C4 dure environ 6 secondes.
- B La commande C2 dure environ 2 seconde.
- C La commande C5 dure environ 4 secondes.
- D La commande C2 dure environ 3 seconde.
- E La commande C1 dure environ 2 secondes.
- F La commande C3 dure environ 2 secondes.
- G La commande C6 dure environ 7 secondes.
- H La commande C4 dure environ 4 secondes.

**Question 21 ♣** La sortie de « `echo "$PATH"` » est « `/usr/bin:/bin` ». La sortie de « `type rm` » est « `rm est /usr/bin/rm` ». « `/usr/bin/rm` » et « `/bin/sh` » sont conforme au standard POSIX. Vous avez tous les droits d'accès sur « `/a` » et « `/a/rm` » et le script « `/a/rm` » a le contenu suivant :

```
#!/bin/sh
echo "rm"
```

Lesquels des énoncés suivants sont vrais ?

- A La commande « `PATH="/a:/usr/bin:/bin" rm rm` » affiche exactement le texte « `rm` ».
- B La commande « `export PATH="/a"; rm` » affiche exactement le texte « `rm` ».
- C La commande « `rm rm` » affiche exactement le texte « `rm` ».
- D La commande « `PATH="/a/rm" rm rm` » affiche exactement le texte « `rm` ».
- E La commande « `cd /a; rm rm` » affiche exactement le texte « `rm` ».
- F La commande « `/a/rm rm` » affiche exactement le texte « `rm` ».
- G La commande « `/bin/sh /a/rm` » affiche exactement le texte « `rm` ».

**Question 22 ♣** Parmi les commandes suivantes, lesquelles produisent systématiquement la sortie suivante ?

```
abc
def
```

A { echo def; echo abc; } | tac

B { echo def; echo abc; } > tac

C echo def | echo abc | tac

D ( echo def; echo abc; ) | tac

E echo def; echo abc; | tac

F echo abc | tac; echo def | tac

**Question 23 ♣** Vous avez tous les droits d'accès sur le répertoire « /final » et son fichier « /final/banane ». Lesquels des énoncés suivants sont vrais ?

A « rm /final/banane || echo "banane" » affiche le texte « banane ».

B « rm /final/banane && echo "banane" » supprime le fichier « /final/banane ».

C « ( exit 1 ) && echo "banane" » affiche le texte « banane ».

D « rm /final/banane && echo "banane" » affiche le texte « banane ».

E « ( exit 0 ) || echo "banane" » affiche le texte « banane ».

F « rm /final/banane || echo "banane" » supprime le fichier « /final/banane ».

## 5 Scriptage

**Question 24** Soit la mise en situation suivante. Quelle sera la sortie produite par « ./test.sh » à la fin ?

```
rak@koios:~$ ls -l
total 5
-rwxrwxr-x 1 rak rak 134 2025-04-13 20:18 test.sh
rak@koios:~$ cat test.sh
#!/bin/sh

if echo "test" | grep t* > /dev/null; then
    echo "1"
else
    echo "2"
fi

if echo "abcd" | grep t* > /dev/null; then
    echo "3"
else
    echo "4"
fi
rak@koios:~$ ./test.sh
```

A 1  
3

B 2  
3

C 1  
4

D 2  
4

**Question 25** Soit le script « ./mystere » suivant :

```
#!/bin/bash
x=0
y=3
f() {
    local x=$((x + 1))
    y=$((y - 1))
}
for a in "$@"; do f; done
echo "x:$x y:$y"
```

Quelle sera la sortie de « ./mystere a b c » ?

A x:0 y:0

B x:3 y:0

C x:3 y:3

D x:0 y:3

**Question 26** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
c=""
for f in pomme banane ananas; do
    if [ "$f" = "ananas" ]; then
        exit 1
    fi
    c="$f $c"
done
echo "$c"
```

A pomme banane ananas

B 1

C Aucune sortie

D ananas banane pomme

**Question 27 ♣** Vous avez toutes les permissions sur le répertoire courant. Le script « ./map » a les permissions 555 et le contenu suivant :

```
#!/bin/sh
F="$1"
shift
for x in "$@"; do
    $F "$x"
done
```

Parmi les énoncés suivants, lesquels sont vrais ?

A « ./map echo "Bonjour INF1070" » produit la ligne « Bonjour INF1070 ».

B « ./map "echo Bonjour INF1070" » produit, entre autres, la ligne « Bonjour ».

C « ./map head \* » produit, entre autres, la ligne « #!/bin/sh ».

D « ./map echo "Bonjour INF1070" » produit une ligne, avec seulement « Bonjour ».

E « ./map echo a b c » produit, entre autres, la ligne « a ».

**Question 28** Soient le script shell « `./script1` » suivant :

```
#!/bin/sh
exit 123
```

et le script shell « `./script2` » suivant :

```
#!/bin/sh
if ./script1; then
    echo "$?"
else
    echo "$?"
    echo "$?"
fi
```

Les deux fichiers sont exécutables. Quelle sera la sortie de « `./script2` » ?

- |                                     |                                       |  |
|-------------------------------------|---------------------------------------|--|
| <input type="checkbox"/> A 123      | <input type="checkbox"/> C 0          | <input type="checkbox"/> E Aucune sortie |
| <input type="checkbox"/> B 123<br>0 | <input type="checkbox"/> D 123<br>123 |  |

## 6 Réseau

**Question 29** Quel protocole se charge de router des paquets de données entre interfaces réseaux de machines connectées à internet ?

- |  |  |
|--|--|
| <input type="checkbox"/> A Le protocole SSH  | <input type="checkbox"/> D Le protocole ADDR |
| <input type="checkbox"/> B Le protocole IP   | <input type="checkbox"/> E Le protocole HTTP |
| <input type="checkbox"/> C Le protocole DHCP | <input type="checkbox"/> F Le protocole DNS  |

**Question 30** Si je visite le site `https://inf1070.uqam.ca`, Firefox m'affiche un cadenas à côté de l'URL dans le navigateur. Cela signifie que :

- |  |  |
|--|--|
| <input type="checkbox"/> A La page ne sera jamais modifiée                                   | <input type="checkbox"/> C La page n'est pas modifiable par l'utilisateur        |
| <input type="checkbox"/> B Il faut entrer un mot de passe pour accéder au contenu de la page | <input type="checkbox"/> D La connexion est sécurisée au travers d'un certificat |
|  | <input type="checkbox"/> E La connexion s'est faite au travers du protocole DNS  |

## Annexe détachable

**bg** [*job\_id*] Exécute une tâche en arrière plan

**cat** Concaténer des fichiers et les afficher sur la sortie standard. Si aucun fichier n'est fourni, ou si le fichier est « - », **cat** lit l'entrée standard.

**cd** [*rep*] Change le répertoire de travail en cours par *rep*.

### Descripteurs de fichiers

1 = sortie standard, 2 = sortie standard d'erreur

**echo** Affiche une ligne de texte.

**fg** [*job\_id*] Exécute une tâche en avant plan

**exit** Termine le shell avec un code de retour. Si omis, le code de retour est celui de la dernière commande exécutée.

**for** NOM in MOTS ...; do COMMANDES; done

Exécute des commandes pour chaque membre d'une liste. Pour chaque élément dans MOTS, NOM est défini à cet élément, et les COMMANDES sont exécutées. Le code de sortie est celui la dernière commande exécutée.

**grep** Recherche les lignes correspondant à un motif donné. Par défaut, **grep** affiche les lignes qui contiennent une correspondance au motif. Code de retour : 0 si une ligne a été trouvée, 1 sinon.

**head** Afficher le début du contenu de fichiers.

**if** Exécute des commandes selon une condition.

Usage : **if** *lst1*; **then** *lst2*; [ **else** *lst3*; ] **fi**

La liste « *lst1* » est exécutée. Si elle se termine avec le code zéro, alors la liste « *lst2* » est exécutée. Sinon, la liste « *lst3* » est exécutée si elle existe. Le code de retour de l'ensemble est celui de la dernière commande exécutée ou zéro si aucune condition n'était vraie.

**jobs** Affiche les tâches qui sont exécutées.

-1 : donne plus d'information sur les tâches, incluant le numéro de la tâche et son PID

**kill** Envoie un signal à des processus. Les processus sont

identifié par leurs PID ou par des spécifications de tâches (préfixées par %). Si on ne précise pas de signal, TERM est envoyé, qui demande au processus de se terminer.

**ls** Afficher le contenu de répertoires.

-1 : utiliser un format d'affichage long.

**PATH** Variable utilisée par le shell

Elle contient la liste ordonnée des répertoires où le shell va chercher ses commandes avoir vérifié ses primitives.

### Paramètres

**\$#** : le nombre de paramètres positionnels.

**\$@** : l'ensemble des paramètres positionnels sous forme de liste.

**ps** Affiche un instantané des processus en cours. Options :

-e : affiche tous les processus

-f : liste plus de colonnes

-j : liste encore plus de colonnes

**pwd** Afficher le nom du répertoire de travail actuel.

**rm** Effacer des fichiers et des répertoires. Code de retour : 0 en cas de succès, 1 en cas d'échec.

**sed** Éditeur de flux pour filtrer et transformer du texte.

-E utilise des expressions régulières étendues.

**shift** Décale la liste des arguments d'une position vers la gauche.

**sleep** Attendre une durée déterminée (en secondes).

**tac** Concatène et affiche des fichiers à l'envers (la dernière ligne en premier).

**test**, [

Teste l'expression entre crochets [ ] et renvoie un état de sortie correspondant au résultat de la comparaison (0 pour vrai, 1 pour faux).

**arg1 = arg2** : Teste l'égalité entre deux chaînes de caractères.





Remplissez complètement la case (■) pour indiquer une bonne réponse (pas de ×, pas de ✓, pas de cercles).  
Les réponses aux questions sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte.

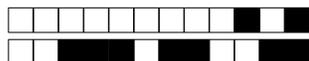
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

← Codez les 8 chiffres de votre code permanent ci-contre, et inscrivez-le à nouveau ci-dessous avec vos nom et prénom. **Les copies sans code permanent ne seront pas corrigées.**

Code permanent :
.....
Prénom :
.....
Nom :
.....

- Question 1 :  A  B  C  D  E  F
- Question 2 :  A  B  C  D  E  F  G  H
- Question 3 :  A  B  C  D
- Question 4 :  A  B  C  D  E  F  G  H
- Question 5 :  A  B  C  D  E  F  G  H
- Question 6 :  A  B  C  D  E
- Question 7 :  A  B  C  D  E
- Question 8 :  A  B  C  D  E  F
- Question 9 :  A  B  C  D  E  F
- Question 10 :  A  B  C  D
- Question 11 :  A  B  C  D  E
- Question 12 :  A  B  C  D  E  F  G  H  I
- Question 13 :  A  B  C  D  E  F  G  H  I
- Question 14 :  A  B  C  D
- Question 15 :  A  B  C  D  E  F  G

- Question 16 :  A  B  C  D  E  F
- Question 17 :  A  B  C  D  E  F  G
- Question 18 :  A  B  C  D
- Question 19 :  A  B  C  D  E  F
- Question 20 :  A  B  C  D  E  F  G  H
- Question 21 :  A  B  C  D  E  F  G
- Question 22 :  A  B  C  D  E  F
- Question 23 :  A  B  C  D  E  F
- Question 24 :  A  B  C  D
- Question 25 :  A  B  C  D
- Question 26 :  A  B  C  D
- Question 27 :  A  B  C  D  E
- Question 28 :  A  B  C  D  E
- Question 29 :  A  B  C  D  E  F
- Question 30 :  A  B  C  D  E



+5/14/51+

INF1070 Utilisation et administration des systèmes informatiques  
Examen Final — Hiver 2025  
Samedi 26 avril — Durée 3 heures  
Ryan Kavanagh et Quentin Stiévenart

- Répondez sur le formulaire (dernière page). Seulement la dernière page sera prise en compte pour la correction.
- Notez aussi votre nom ci-dessous.
- Aucun document ni appareil électronique n'est autorisé.
- L'annexe est détachable.
- Les questions faisant apparaître le symbole ♣ peuvent avoir plusieurs bonnes réponses. Les autres ont une unique bonne réponse.
- Chacune des questions recevra un score entre 0 et 5 points et toutes les questions ont la même pondération.
- Les mauvaises réponses sont pénalisées. Le score d'une question ne peut pas devenir négatif avec les pénalités.
- Le formulaire (dernière page) est lié à votre copie. L'ordre des réponses est différent d'une copie à l'autre. Il ne faut donc pas mélanger le formulaire de réponse de deux copies différentes.

Code permanent :

.....

Prénom :

.....

Nom :

.....

# 1 Expressions régulières

**Question 1 ♣** Les codes postaux canadiens sont des codes alphanumériques formés de six caractères avec le format « LNL NLN », où L est une lettre entre A et Z et N est un chiffre entre 0 et 9. Il y a un espace entre les deux groupes (« LNL », espace, « NLN »). Par exemple, « HOH OH0 » et « P5Q 5P7 » sont des codes postaux, mais « A3B4C6 », « AA5 33B » et « Z5 53H » ne le sont pas. Lesquelles des ERE suivantes ont **une correspondance** avec **au moins** un code postal valide ?

- |   |   |
|---|---|
| <input type="checkbox"/> A <code>^[A-Z][0-9][A-Z][0-9][A-Z][0-9]\$</code> | <input type="checkbox"/> D <code>(.)(.)(.)\1\2\3</code>     |
| <input type="checkbox"/> B <code>[\^0-9][\^A-Z][\^0-9]</code>             | <input type="checkbox"/> E <code>(.[0-9].\$ ^[A-Z].)</code> |
| <input type="checkbox"/> C <code>([A-Z][0-9]{2,})</code>                  | <input type="checkbox"/> F <code>([A-Z]?[0-9]){3}</code>    |

**Question 2 ♣** Une expression régulière  $R_1$  est **équivalente** à une expression régulière  $R_2$  si, pour toute chaîne de caractères  $C$ ,  $R_1$  a une correspondance avec  $C$  si et seulement si  $R_2$  a une correspondance avec  $C$ . Concrètement, cela veut dire que, pour toute entrée, `grep 'R1'` et `grep 'R2'` sélectionneront les mêmes lignes. Indiquez toutes les paires d'expressions régulières étendues (ERE) équivalentes :

- |   |  |
|---|--|
| <input type="checkbox"/> A <code>(a+)?</code> et <code>a*</code>          | <input type="checkbox"/> E <code>a?(b c)*</code> et <code>(a?b a?c)*</code>      |
| <input type="checkbox"/> B <code>(a?)+</code> et <code>a*</code>          | <input type="checkbox"/> F <code>a(b[cd] [bc]d)</code> et <code>a(bc bc)d</code> |
| <input type="checkbox"/> C <code>a(b c)d</code> et <code>(abd acd)</code> | <input type="checkbox"/> G <code>[\^abc\$]</code> et <code>[\^abc]\$</code>      |
| <input type="checkbox"/> D <code>[\^.*\$]</code> et <code>[\^.*]</code>   | <input type="checkbox"/> H <code>([ab])[ab]</code> et <code>([ab])\1</code>      |

**Question 3 ♣** Chaque ligne du fichier `courriels.html` est un lien vers une URL avec le schéma `mailto`, comme suit :

```
<a href="mailto:alphonse@desjardins.com">Alphonse Desjardins</a>
<a href="mailto:joseph-armand@bombardier.com">Joseph-Armand "Jo" Bombardier</a>
<a href="mailto:mary@la-bolduc.ca" rel="external">Mary Travers</a>
<a href="mailto:paul@manouvellemaison.ca">Paul de Chomedey de Maisonneuve</a>
```

Parmi les commandes suivantes, lesquelles permettent d'extraire de ce fichier toutes les adresses courriels et seulement celles-ci, sans les guillemets ?

- |  |
|--|
| <input type="checkbox"/> A <code>sed -E 's/.*mailto:([\^"]+).*/\1/' courriels.html</code>        |
| <input type="checkbox"/> B <code>sed -E 's/.*mailto:(.*)"/\1/' courriels.html</code>             |
| <input type="checkbox"/> C <code>sed -E 's/.*mailto://' courriels.html   sed -E 's/".*//'</code> |
| <input type="checkbox"/> D <code>sed -E 's/".*//' courriels.html   sed -E 's/.*://'</code>       |

**Question 4 ♣** Lesquelles des URLs suivantes ont une correspondance avec l'ERE « `((f|ht)tps?://|mailto:)(infor?|ma|thé)+(tiques?)?.uqam.ca` » ?

- |  |  |
|--|--|
| <input type="checkbox"/> A <code>://ma.uqam.ca</code>                | <input type="checkbox"/> E <code>mailto:thé@uqam.ca</code>           |
| <input type="checkbox"/> B <code>https://informatique.uqam.ca</code> | <input type="checkbox"/> F <code>ftp://info.uqam.ca</code>           |
| <input type="checkbox"/> C <code>mailto:thétiques?.uqam.ca</code>    | <input type="checkbox"/> G <code>https://uqam.ca</code>              |
| <input type="checkbox"/> D <code>HTTP://MA.UQAM.CA</code>            | <input type="checkbox"/> H <code>http://mathématiques.uqam.ca</code> |

**Question 5 ♣** Lesquelles des ERE suivantes ont une correspondance avec la chaîne « `épaississait` » ?

- |  |  |
|--|--|
| <input type="checkbox"/> A <code>(..)(..)\2\1</code> | <input type="checkbox"/> E <code>(.)\1\1</code>  |
| <input type="checkbox"/> B <code>((..)\2</code>      | <input type="checkbox"/> F <code>^(.)*\1</code>  |
| <input type="checkbox"/> C <code>(..)*\1</code>      | <input type="checkbox"/> G <code>(is).+\1</code> |
| <input type="checkbox"/> D <code>(..)(..)\2\1</code> | <input type="checkbox"/> H <code>((..)\1</code>  |

## 2 Administration

**Question 6 ♣** Parmi les éléments suivants, lesquels font partie des tâches d'un gestionnaire de paquets ?

- A Installer les dépendances d'un paquet
- B Compter les paquets réseau
- C Installer des fichiers exécutables
- D Surveiller en temps réel l'utilisation CPU et RAM des logiciels installés
- E Désinstaller un paquet

**Question 7 ♣** Parmi les énoncés suivants, lesquels sont forcément vrais pour un démon (daemon) ?

- A Il peut être lancé automatiquement au démarrage du système
- B Il est automatiquement fermé dès que l'utilisateur ferme son terminal
- C Il s'exécute en arrière plan
- D Il s'exécute en avant plan et interagit directement avec l'utilisateur
- E Il s'exécute avec une priorité élevée

**Question 8 ♣** Parmi les énoncés suivants, lesquels sont vrais ?

- A Les journaux des services sont généralement stockés dans `/var/log/`
- B Les journaux des services sont généralement stockés dans le dossier utilisateur
- C Les fichiers de journalisation peuvent être généralement analysés avec des outils comme `grep` pour rechercher des événements spécifiques
- D Les fichiers de journalisation contiennent des nouvelles importantes sur les nouveaux paquets logiciels disponibles
- E Certains fichiers de journalisation peuvent être automatiquement archivés
- F Les journaux des services sont généralement stockés dans `/etc/log/`

### 3 Processus

Les 4 questions suivantes se font dans le contexte suivant.

```
$ ps -efj
UID      PID  PPID  PGID   SID  C  STIME TTY      TIME CMD
root      1    0     1     1  0  11:33 ?        00:00:00 /sbin/init
root     507    1   507   507  0  11:33 ?        00:00:00 /usr/sbin/anacron -d -q -s
avahi    508    1   508   508  0  11:33 ?        00:00:00 avahi-daemon: running [debian.local]
root     509    1   509   509  0  11:33 ?        00:00:00 /usr/sbin/cron -f
message+ 510    1   510   510  0  11:33 ?        00:00:00 /usr/bin/dbus-daemon --system --address=sys
root     512    1   512   512  0  11:33 ?        00:00:00 /usr/libexec/low-memory-monitor
debian  1845  1768  1845  1768  0  11:34 pts/0    00:00:00 tmux
debian  1847    1  1847  1847  0  11:34 ?        00:00:00 tmux
debian  1848  1847  1848  1848  0  11:34 pts/1    00:00:00 -bash
debian  1858  1848  1858  1848  0  11:35 pts/1    00:00:00 sleep 1200
debian  1859  1847  1859  1859  0  11:35 pts/2    00:00:00 -bash
debian  1865  1859  1865  1859  1  11:35 pts/2    00:00:06 htop
debian  1866  1847  1866  1866  0  11:35 pts/3    00:00:00 -bash
debian  1951  1847  1951  1951  0  11:41 pts/4    00:00:00 -bash
debian  1957  1951  1957  1951  0  11:41 pts/4    00:00:00 vim
debian  1959  1957  1957  1951  0  11:41 pts/4    00:00:00 cat
debian  1960  1866  1960  1866  0  11:41 pts/3    00:00:00 ps -efj
```

**Question 9 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |  |   |
|--|---|
| <input type="checkbox"/> A L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code>   | <input type="checkbox"/> D L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>sleep</code>     |
| <input type="checkbox"/> B L'utilisateur <code>debian</code> a le droit d'envoyer un signal au processus <code>avahi-daemon</code> | <input type="checkbox"/> E L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>dbus-daemon</code> |
| <input type="checkbox"/> C L'utilisateur <code>root</code> a le droit d'envoyer un signal au processus <code>sleep</code>          | <input type="checkbox"/> F L'utilisateur <code>avahi</code> a le droit d'envoyer un signal au processus <code>sleep</code>      |

**Question 10** Parmi les affirmations suivantes, laquelle est vraie pour le contexte donné ?

- |   |  |
|---|--|
| <input type="checkbox"/> A L'identifiant de processus de la commande <code>sleep</code> est 1848  | <input type="checkbox"/> C L'identifiant de processus de la commande <code>sleep</code> est <code>pts/1</code> |
| <input type="checkbox"/> B L'identifiant de processus de la commande <code>sleep</code> est 11:35 | <input type="checkbox"/> D L'identifiant de processus de la commande <code>sleep</code> est 1858               |

**Question 11 ♣** Parmi les affirmations suivantes, lesquelles sont vraies pour le contexte donné ?

- |   |  |
|---|--|
| <input type="checkbox"/> A la commande <code>cat</code> a été lancée directement depuis <code>bash</code>     | <input type="checkbox"/> D la commande <code>cat</code> a été lancée directement depuis <code>tmux</code>    |
| <input type="checkbox"/> B La commande <code>cat</code> a été lancée directement depuis <code>vim</code>      | <input type="checkbox"/> E la commande <code>ps -efj</code> a été lancée directement depuis <code>vim</code> |
| <input type="checkbox"/> C la commande <code>ps -efj</code> a été lancée directement depuis <code>bash</code> |  |

Pour les questions suivantes, on a le contexte suivant.

```
$ jobs -l
[1] 3532 Running gnome-calculator &
[2] 3643 Running nautilus &
[3] 3710 Running xeyes &
```

**Question 12 ♣** Après avoir effectué avec succès la commande `kill 3643`, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>xeyes</code> est terminé                        | <input type="checkbox"/> F Le processus <code>nautilus</code> s'exécute en avant-plan           |
| <input type="checkbox"/> B Le processus <code>nautilus</code> s'exécute en arrière-plan       | <input type="checkbox"/> G Le processus <code>xeyes</code> s'exécute en arrière-plan            |
| <input type="checkbox"/> C Le processus <code>xeyes</code> s'exécute en avant-plan            | <input type="checkbox"/> H Le processus <code>gnome-calculator</code> s'exécute en arrière-plan |
| <input type="checkbox"/> D Le processus <code>gnome-calculator</code> s'exécute en avant-plan | <input type="checkbox"/> I Le processus <code>gnome-calculator</code> est terminé               |
| <input type="checkbox"/> E Le processus <code>nautilus</code> est terminé                     |   |

**Question 13 ♣** À la place de la commande de la question précédente, si j'avais effectué la commande `fg %3` et ensuite entré `^C` (Contrôle-C) au clavier, quelles affirmations parmi les suivantes seront vraies ?

- |   |   |
|---|---|
| <input type="checkbox"/> A Le processus <code>gnome-calculator</code> s'exécute en arrière-plan | <input type="checkbox"/> E Le processus <code>gnome-calculator</code> est terminé     |
| <input type="checkbox"/> B Le processus <code>xeyes</code> est terminé                          | <input type="checkbox"/> F Le processus <code>xeyes</code> s'exécute en avant-plan    |
| <input type="checkbox"/> C Le processus <code>gnome-calculator</code> s'exécute en avant-plan   | <input type="checkbox"/> G Le processus <code>nautilus</code> est terminé             |
| <input type="checkbox"/> D Le processus <code>nautilus</code> s'exécute en arrière-plan         | <input type="checkbox"/> H Le processus <code>nautilus</code> s'exécute en avant-plan |
|   | <input type="checkbox"/> I Le processus <code>xeyes</code> s'exécute en arrière-plan  |

## 4 Shell avancé

**Question 14** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
x="a"
y="b"
z="c      d"
echo "$x" '$y' $z
```

- |   |                                      |  |                                      |
|---|--------------------------------------|--|--------------------------------------|
| <input type="checkbox"/> A a \$y c      d | <input type="checkbox"/> B a \$y c d | <input type="checkbox"/> C \$x \$y \$z | <input type="checkbox"/> D \$x y c d |
|---|--------------------------------------|--|--------------------------------------|

**Question 15 ♣** Soit le script shell suivant :

```
#!/bin/sh
```

```
x=9
x="9*$x"
echo "$x"
```

```
y=0
y="$(echo "$y")"
echo "$y"
```

```
z=1
z="$(echo '$z ')"
echo "$z"
```

Quelles lignes apparaissent, entre autres, dans sa sortie ?

- A 81       B \$y       C 0       D \$x       E \$z       F 1       G 9\*9

**Question 16 ♣** Le répertoire courant contient le fichier « Mon rapport.txt ». Vous avez tous les droits d'accès sur le répertoire courant et ses fichiers. Lesquels des énoncés suivants sont vrais ?

- A La commande « a=Mon rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 B La commande « a=Mon\ rapport.txt; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 C La commande « a="Mon\*.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 D La commande « a="Mon rapport.txt"; rm "\$a" » supprime le fichier « Mon rapport.txt »  
 E La commande « a="Mon rapport.txt"; rm \$a » supprime le fichier « Mon rapport.txt »  
 F La commande « a="Mon"; rm "\$a\*" » supprime le fichier « Mon rapport.txt »

**Question 17** Le script shell suivant :

```
#!/bin/sh
```

```
cd /bin
pwd > /tmp/pwd.log
( cd /etc; pwd; )
pwd >> /tmp/pwd.log
{ cd /usr; pwd; }
pwd >> /tmp/pwd.log
{ cd /tmp; pwd; } | rev
pwd >> /tmp/pwd.log
```

a la sortie suivante :

```
/etc
/usr
pmt/
```

Quelle sera le contenu du fichier « /tmp/pwd.log » ?

- A /bin       B /bin       C /bin       D /bin       E /bin       F /bin       G /bin  
 /bin       /etc       /bin       /etc       /bin       /bin       /bin  
 /bin       /etc       /bin       /usr       /usr       /bin       /usr  
 /usr       /tmp       /tmp       /usr       /tmp       /bin       /usr

**Question 18 ♣** La commande « `cmd` » présente dans le PATH a toujours le même comportement et produit la sortie suivante dans le terminal :

```
abc
xbz
```

Parmi les énoncés suivants, lesquels sont forcément vrais ?

- A Il est possible que la conduite « `cmd | grep q` » affiche la ligne « `abc` » dans le terminal.
- B Si « `cmd > /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.
- C Si « `cmd | grep q` » n’affiche aucune ligne, alors « `cmd > /dev/null` » n’affiche rien dans le terminal.
- D Si « `cmd 2> /dev/null` » n’affiche rien dans le terminal, alors « `cmd | grep q` » affichera deux lignes.

**Question 19 ♣** Vous avez tous les droits d’accès sur le répertoire courant qui est vide. La commande « `oui` » produit toujours le texte « `pomme` » sur sa sortie standard. La commande « `non` » produit toujours le texte « `pomme` » sur sa sortie d’erreur. Suites à quelles commandes le fichier « `fruits` » contiendra-t-il le mot « `pomme` » ?

- |  |   |   |
|--|---|---|
| <input type="checkbox"/> A <code>oui &gt; fruits &gt;&amp;2</code>   | <input type="checkbox"/> C <code>oui &gt; fruits 2&gt;&amp;1</code> | <input type="checkbox"/> E <code>oui &gt;&amp;2 2&gt; fruits</code> |
| <input type="checkbox"/> B <code>non 2&gt; fruits 1&gt;&amp;2</code> | <input type="checkbox"/> D <code>non 2&gt;&amp;1 &gt; fruits</code> | <input type="checkbox"/> F <code>non &gt; fruits 2&gt;&amp;1</code> |

**Question 20 ♣** Soit les commandes suivantes, où chaque commande est identifiée par un commentaire :

```
sleep 1; sleep 1           # C1
sleep 1 || sleep 2        # C2
sleep 1 | sleep 2         # C3
{ sleep 2; sleep 2; } | sleep 2 # C4
sleep 2 && sleep 2         # C5
sleep 3 & sleep 4 &       # C6
```

Lesquels des énoncés suivants sont vrais ?

- |  |  |
|--|--|
| <input type="checkbox"/> A La commande C1 dure environ 2 secondes. | <input type="checkbox"/> E La commande C4 dure environ 4 secondes. |
| <input type="checkbox"/> B La commande C3 dure environ 2 secondes. | <input type="checkbox"/> F La commande C6 dure environ 7 secondes. |
| <input type="checkbox"/> C La commande C4 dure environ 6 secondes. | <input type="checkbox"/> G La commande C5 dure environ 4 secondes. |
| <input type="checkbox"/> D La commande C2 dure environ 3 seconde.  | <input type="checkbox"/> H La commande C2 dure environ 2 seconde.  |

**Question 21 ♣** La sortie de « `echo "$PATH"` » est « `/usr/bin:/bin` ». La sortie de « `type rm` » est « `rm est /usr/bin/rm` ». « `/usr/bin/rm` » et « `/bin/sh` » sont conforme au standard POSIX. Vous avez tous les droits d’accès sur « `/a` » et « `/a/rm` » et le script « `/a/rm` » a le contenu suivant :

```
#!/bin/sh
echo "rm"
```

Lesquels des énoncés suivants sont vrais ?

- A La commande « `/a/rm rm` » affiche exactement le texte « `rm` ».
- B La commande « `rm rm` » affiche exactement le texte « `rm` ».
- C La commande « `export PATH="/a"; rm` » affiche exactement le texte « `rm` ».
- D La commande « `/bin/sh /a/rm` » affiche exactement le texte « `rm` ».
- E La commande « `PATH="/a/rm" rm rm` » affiche exactement le texte « `rm` ».
- F La commande « `cd /a; rm rm` » affiche exactement le texte « `rm` ».
- G La commande « `PATH="/a:/usr/bin:/bin" rm rm` » affiche exactement le texte « `rm` ».

**Question 22 ♣** Parmi les commandes suivantes, lesquelles produisent systématiquement la sortie suivante ?

```
abc
def
```

- |   |  |
|---|--|
| <input type="checkbox"/> A { echo def; echo abc; } > tac  | <input type="checkbox"/> D echo def   echo abc   tac     |
| <input type="checkbox"/> B ( echo def; echo abc; )   tac  | <input type="checkbox"/> E echo def; echo abc;   tac     |
| <input type="checkbox"/> C echo abc   tac; echo def   tac | <input type="checkbox"/> F { echo def; echo abc; }   tac |

**Question 23 ♣** Vous avez tous les droits d'accès sur le répertoire « /final » et son fichier « /final/banane ». Lesquels des énoncés suivants sont vrais ?

- |   |   |
|---|---|
| <input type="checkbox"/> A « ( exit 1 ) && echo "banane" » affiche le texte « banane ».                 | <input type="checkbox"/> D « ( exit 0 )    echo "banane" » affiche le texte « banane ».       |
| <input type="checkbox"/> B « rm /final/banane && echo "banane" » supprime le fichier « /final/banane ». | <input type="checkbox"/> E « rm /final/banane    echo "banane" » affiche le texte « banane ». |
| <input type="checkbox"/> C « rm /final/banane    echo "banane" » supprime le fichier « /final/banane ». | <input type="checkbox"/> F « rm /final/banane && echo "banane" » affiche le texte « banane ». |

## 5 Scriptage

**Question 24** Soit la mise en situation suivante. Quelle sera la sortie produite par « ./test.sh » à la fin ?

```
rak@koios:~$ ls -l
total 5
-rwxrwxr-x 1 rak rak 134 2025-04-13 20:18 test.sh
rak@koios:~$ cat test.sh
#!/bin/sh

if echo "test" | grep t* > /dev/null; then
    echo "1"
else
    echo "2"
fi

if echo "abcd" | grep t* > /dev/null; then
    echo "3"
else
    echo "4"
fi
rak@koios:~$ ./test.sh
```

- |                                   |                                   |                                   |                                   |
|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| <input type="checkbox"/> A 2<br>4 | <input type="checkbox"/> B 1<br>4 | <input type="checkbox"/> C 1<br>3 | <input type="checkbox"/> D 2<br>3 |
|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|

**Question 25** Soit le script « ./mystere » suivant :

```
#!/bin/bash
x=0
y=3
f() {
    local x=$((x + 1))
    y=$((y - 1))
}
for a in "$@"; do f; done
echo "x:$x y:$y"
```

Quelle sera la sortie de « ./mystere a b c » ?

A x:0 y:0

B x:3 y:3

C x:3 y:0

D x:0 y:3

**Question 26** Quelle sera la sortie du script suivant ?

```
#!/bin/sh
c=""
for f in pomme banane ananas; do
    if [ "$f" = "ananas" ]; then
        exit 1
    fi
    c="$f $c"
done
echo "$c"
```

A Aucune sortie

B 1

C pomme banane ananas

D ananas banane pomme

**Question 27 ♣** Vous avez toutes les permissions sur le répertoire courant. Le script « ./map » a les permissions 555 et le contenu suivant :

```
#!/bin/sh
F="$1"
shift
for x in "$@"; do
    $F "$x"
done
```

Parmi les énoncés suivants, lesquels sont vrais ?

A « ./map echo a b c » produit, entre autres, la ligne « a ».

B « ./map "echo Bonjour INF1070" » produit, entre autres, la ligne « Bonjour ».

C « ./map echo "Bonjour INF1070" » produit la ligne « Bonjour INF1070 ».

D « ./map head \* » produit, entre autres, la ligne « #!/bin/sh ».

E « ./map echo "Bonjour INF1070" » produit une ligne, avec seulement « Bonjour ».

**Question 28** Soient le script shell « `./script1` » suivant :

```
#!/bin/sh
exit 123
```

et le script shell « `./script2` » suivant :

```
#!/bin/sh
if ./script1; then
    echo "$?"
else
    echo "$?"
    echo "$?"
fi
```

Les deux fichiers sont exécutables. Quelle sera la sortie de « `./script2` » ?

- A 123  
123
- B 123  
0
- C 123  
Aucune sortie
- D 0

## 6 Réseau

**Question 29** Quel protocole se charge de router des paquets de données entre interfaces réseaux de machines connectées à internet ?

- A Le protocole DNS
- B Le protocole DHCP
- C Le protocole IP
- D Le protocole ADDR
- E Le protocole HTTP
- F Le protocole SSH

**Question 30** Si je visite le site `https://inf1070.uqam.ca`, Firefox m'affiche un cadenas à côté de l'URL dans le navigateur. Cela signifie que :

- A La page ne sera jamais modifiée
- B Il faut entrer un mot de passe pour accéder au contenu de la page
- C La connexion s'est faite au travers du protocole DNS
- D La connexion est sécurisée au travers d'un certificat
- E La page n'est pas modifiable par l'utilisateur

## Annexe détachable

**bg** [*job\_id*] Exécute une tâche en arrière plan

**cat** Concaténer des fichiers et les afficher sur la sortie standard. Si aucun fichier n'est fourni, ou si le fichier est « - », **cat** lit l'entrée standard.

**cd** [*rep*] Change le répertoire de travail en cours par *rep*.

### Descripteurs de fichiers

1 = sortie standard, 2 = sortie standard d'erreur

**echo** Affiche une ligne de texte.

**fg** [*job\_id*] Exécute une tâche en avant plan

**exit** Termine le shell avec un code de retour. Si omis, le code de retour est celui de la dernière commande exécutée.

**for** NOM in MOTS ...; do COMMANDES; done

Exécute des commandes pour chaque membre d'une liste. Pour chaque élément dans MOTS, NOM est défini à cet élément, et les COMMANDES sont exécutées. Le code de sortie est celui la dernière commande exécutée.

**grep** Recherche les lignes correspondant à un motif donné. Par défaut, **grep** affiche les lignes qui contiennent une correspondance au motif. Code de retour : 0 si une ligne a été trouvée, 1 sinon.

**head** Afficher le début du contenu de fichiers.

**if** Exécute des commandes selon une condition.

Usage : **if** *lst1*; **then** *lst2*; [ **else** *lst3*; ] **fi**

La liste « *lst1* » est exécutée. Si elle se termine avec le code zéro, alors la liste « *lst2* » est exécutée. Sinon, la liste « *lst3* » est exécutée si elle existe. Le code de retour de l'ensemble est celui de la dernière commande exécutée ou zéro si aucune condition n'était vraie.

**jobs** Affiche les tâches qui sont exécutées.

-1 : donne plus d'information sur les tâches, incluant le numéro de la tâche et son PID

**kill** Envoie un signal à des processus. Les processus sont

identifié par leurs PID ou par des spécifications de tâches (préfixées par %). Si on ne précise pas de signal, TERM est envoyé, qui demande au processus de se terminer.

**ls** Afficher le contenu de répertoires.

-1 : utiliser un format d'affichage long.

**PATH** Variable utilisée par le shell

Elle contient la liste ordonnée des répertoires où le shell va chercher ses commandes avoir vérifié ses primitives.

### Paramètres

**\$#** : le nombre de paramètres positionnels.

**\$@** : l'ensemble des paramètres positionnels sous forme de liste.

**ps** Affiche un instantané des processus en cours. Options :

-e : affiche tous les processus

-f : liste plus de colonnes

-j : liste encore plus de colonnes

**pwd** Afficher le nom du répertoire de travail actuel.

**rm** Effacer des fichiers et des répertoires. Code de retour : 0 en cas de succès, 1 en cas d'échec.

**sed** Éditeur de flux pour filtrer et transformer du texte.

-E utilise des expressions régulières étendues.

**shift** Décale la liste des arguments d'une position vers la gauche.

**sleep** Attendre une durée déterminée (en secondes).

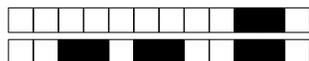
**tac** Concatène et affiche des fichiers à l'envers (la dernière ligne en premier).

**test**, [

Teste l'expression entre crochets [ ] et renvoie un état de sortie correspondant au résultat de la comparaison (0 pour vrai, 1 pour faux).

**arg1 = arg2** : Teste l'égalité entre deux chaînes de caractères.





Remplissez complètement la case (■) pour indiquer une bonne réponse (pas de ×, pas de ✓, pas de cercles).  
Les réponses aux questions sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte.

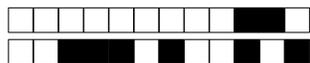
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

← Codez les 8 chiffres de votre code permanent ci-contre, et inscrivez-le à nouveau ci-dessous avec vos nom et prénom. **Les copies sans code permanent ne seront pas corrigées.**

Code permanent :
.....
Prénom :
.....
Nom :
.....

- Question 1 :  A  B  C  D  E  F
- Question 2 :  A  B  C  D  E  F  G  H
- Question 3 :  A  B  C  D
- Question 4 :  A  B  C  D  E  F  G  H
- Question 5 :  A  B  C  D  E  F  G  H
- Question 6 :  A  B  C  D  E
- Question 7 :  A  B  C  D  E
- Question 8 :  A  B  C  D  E  F
- Question 9 :  A  B  C  D  E  F
- Question 10 :  A  B  C  D
- Question 11 :  A  B  C  D  E
- Question 12 :  A  B  C  D  E  F  G  H  I
- Question 13 :  A  B  C  D  E  F  G  H  I
- Question 14 :  A  B  C  D
- Question 15 :  A  B  C  D  E  F  G

- Question 16 :  A  B  C  D  E  F
- Question 17 :  A  B  C  D  E  F  G
- Question 18 :  A  B  C  D
- Question 19 :  A  B  C  D  E  F
- Question 20 :  A  B  C  D  E  F  G  H
- Question 21 :  A  B  C  D  E  F  G
- Question 22 :  A  B  C  D  E  F
- Question 23 :  A  B  C  D  E  F
- Question 24 :  A  B  C  D
- Question 25 :  A  B  C  D
- Question 26 :  A  B  C  D
- Question 27 :  A  B  C  D  E
- Question 28 :  A  B  C  D  E
- Question 29 :  A  B  C  D  E  F
- Question 30 :  A  B  C  D  E



+6/14/37+