



INF1013L ASR1 Unix, examen session 1, 16 mai 2018 14h00-15h00

<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1
<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2
<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4
<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5
<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6
<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7
<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8
<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9

← Indiquez votre code étudiant, de gauche à droite en noircissant une case par colonne.

Nom et prénom

.....

.....

Durée une heure. Aucun document autorisé.
 Remplissez au stylo noir ou bleu la case de l'unique bonne réponse (une croix ne suffit pas).
 N'écrivez rien sur la copie. Ne barrez pas une mauvaise réponse, mettez du blanc.
 Ne redessinez pas une case que vous avez effacée, laissez blanc.
 Cet énoncé ne contient aucune double cotes ("), si vous en voyez, alors ce sont des guillemets (").
 Si vous cochez une mauvaise case, vous perdez 1/(N-1) points, N étant le nombre de cases de la question.

Question 1 Afficher le contenu du fichier «f» sur la sortie standard :

- | | | | |
|----------------------------------|----------------------------------|-----------------------------------|--------------------------------|
| <input type="checkbox"/> f > cat | <input type="checkbox"/> echo <f | <input type="checkbox"/> echo \$f | <input type="checkbox"/> <f |
| <input type="checkbox"/> f cat | <input type="checkbox"/> >f | <input type="checkbox"/> cat >f | <input type="checkbox"/> cat f |

Question 2 Remplacer toutes les séries de plusieurs espaces consécutifs par un seul :

- | | | | |
|---|--|--|---|
| <input type="checkbox"/> sed 's/ */ /' | <input type="checkbox"/> sed 's/\ */ /g' | <input type="checkbox"/> sed 's/[]*/ /' | <input type="checkbox"/> sed 's/ \ */ /' |
| <input type="checkbox"/> sed 's/ */ /g' | <input type="checkbox"/> sed 's/ */ /' | <input type="checkbox"/> sed 's/ */ /g' | <input type="checkbox"/> sed s/\ \ */ /g' |

Question 3 Afficher la ligne lue sur l'entrée standard si elle est égale à la dernière ligne lue :

- for L in < ; if ["\$L" = "\$LL"] ; then echo "\$L" ; else LL="\$L" ; fi ; done
- while read L ; do if ["\$L" = "\$LL"] ; then echo "\$L" ; fi ; LL="\$L" ; done
- while test "\$L" = "\$LL" ; do echo "\$L" ; LL="\$L" ; read L ; done
- while read L ; do ["\$L" = "\$LL"] && echo "\$L" ; LL="\$L" ; done

Question 4 On veut mettre dans «V» le texte «J'ai "faim"». Cochez l'affectation qui NE LE FAIT PAS :

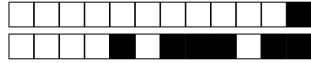
- | | | |
|--|---|---|
| <input type="checkbox"/> V="J'ai \"faim\"" | <input type="checkbox"/> V="J'ai "'faim'" | <input type="checkbox"/> V="J'"ai "faim" |
| <input type="checkbox"/> V=J'"ai' "'faim'" | <input type="checkbox"/> V=J\'ai\ \"faim\"" | <input type="checkbox"/> V='J\'ai "faim"' |

Question 5 Lancer les 2 commandes en parallèle (et non l'une après l'autre) :

- | | | | |
|----------------------------------|------------------------------------|--------------------------------------|---------------------------------|
| <input type="checkbox"/> (a ; b) | <input type="checkbox"/> a ; b | <input type="checkbox"/> (a && bb) & | <input type="checkbox"/> a / b |
| <input type="checkbox"/> a ; b & | <input type="checkbox"/> (a ; b) & | <input type="checkbox"/> a b | <input type="checkbox"/> a && b |

Question 6 Afficher le texte contenu dans la variable «V» :

- | | | | | | |
|----------------------------------|-----------------------------------|---------------------------------|--------------------------------|---------------------------------|-----------------------------------|
| <input type="checkbox"/> cat \$V | <input type="checkbox"/> read \$V | <input type="checkbox"/> read V | <input type="checkbox"/> cat V | <input type="checkbox"/> echo V | <input type="checkbox"/> echo \$V |
|----------------------------------|-----------------------------------|---------------------------------|--------------------------------|---------------------------------|-----------------------------------|



Question 7 Créer un fichier de type texte dont le nom est la date courante :

- echo 2>"\$date"
- echo <date
- echo >date
- mkdir <date
- mkdir \$(date)
- mkdir "\$date"
- echo >"\$DATE"
- echo >"\$(date)"

Question 8 Expression régulière trouvant les lignes commençant et finissant par une étoile :

- [*].*[*]
- '*!*'*
- ^\.*.**\$
- ***
- ^**\$
- *.*
- ^*.*\$
- ^***\$
- [^*].*[*\$]
- ***

Question 9 Afficher le plus grand nombre contenu dans le fichier «A», il y a un nombre par ligne :

- tail -1 \$(sort -n <A)
- sort -n A | tail -1
- sort -n A ; tail -1 A
- A=\$(sort -n A) ; tail -1 \$A

Question 10 Mettre dans le fichier «D» le contenu du fichier «F» sans la première ligne :

- read A B <F;echo "\$B" >D
- read A <F ; cat <F >D
- read A <F ; cat F >D
- (read A ; cat) <F >D
- cp D F ; read A <D
- cat F | read A >D

Question 11 Copier le fichier «f1» sous le nom «f2» :

- cp f1 f2
- cat f1 f2
- cat f2 >f1
- cp <f1 >f2
- f2 = f1

Question 12 Mettre le texte «T» dans la variable «V» :

- T > V
- V = 'T'
- \$V='T'
- V < T
- V= "T"
- V=T

Question 13 Lister les lignes du fichier «F» contenant au moins un «a» et au moins un «b» (par exemple «bzcat») :

- grep '[ab]' <F
- grep '[ab]' F
- grep 'a.*b' F
- grep a <F | grep b
- grep '[ab]*' F
- grep F '(ab)*'
- grep F '[ab]'
- grep a b F

Question 14 Laquelle de ces commandes reçoit deux arguments quand elle s'exécute :

- echo 'e f'
- echo echo z
- echo \$(echo g)
- echo a " " b
- echo x #
- echo '\$"A"
- echo .* .*
- echo c\ d

Question 15 Expression régulière trouvant les lignes commençant par «a» :

- [!a]
- a!
- ^a
- a^
- a\$
- \$a
- [^a]
- !a

Question 16 Nom du fichier exécuté par le shell à chaque fois que vous vous connectez :

- /etc/start
- ~/start
- ~/.profile
- ~/profile
- /etc/login

Question 17 Expression régulière étendue trouvant les échanges de contenu de 2 variables en langage C, par exemple celle-ci : «tmp=x;x=y;y=tmp»

- ([a-z]+)=[a-z]+;\2=[a-z]+;\3=\1
- ([a-z]+)=[a-z]+;\2=[a-z]+;\3=\1
- [a-z]+=[a-z]+;\2=[a-z]+;\3=\1
- [a-z]=[a-z];\1=[a-z];\2=\3



Question 18 Afficher le contenu complet du fichier nommé «F» :

- for L in F ; do echo "\$L" ; done cat >F
 while read L ; do echo "\$L" ; done <F read L <F ; echo "\$L"
 for L in F ; do echo \$L ; done cat "\$F"

Question 19 Expression régulière trouvant toutes les lignes NON vides :

- ^.\$.* [a-z] ^\$? \. .

Question 20 Afficher tous les noms d'entité contenant une lettre suivi d'un chiffre :

- echo *[az][09]* echo * [a-z] [0-9] echo *[a-z][0-9]*
 echo *[a-z0-9] echo [a-z],[0-9] echo [a-z,0-9]*

Question 21 Détruire le répertoire «D» et son contenu :

- D>/dev/null rm -r D rmdir D rm D/* remove D

Question 22 Inverser les 2 premiers mots de chacune des lignes lues sur l'entrée standard :

- sed 's/[^]* [^]*/\2 \1/'
 while read A B C ; do echo "\$B" "\$A" "\$C" ; done
 for A B in < ; do echo "\$B" "\$A" ; done
 while read A B ; do echo "\$B" "\$A" ; done

Question 23 Afficher deux lignes : «(A)» puis «(B)»

- for V in "A B" ; do echo "(\$V)" ; done for V in "(A) (B)" ; do echo "\$V" ; done
 for V in A B ; do echo "(\$V)" ; done for V in [AB] ; do echo "(\$V)" ; done

Question 24 Mettre dans la variable «V» tous les chemins relatifs menant à des entités dont le nom se termine par «~» :

- find . -name *~ | read V find . -name '*~' >\$V V=(echo *~)
 find . -name *~ >V find . -name "*~" | V V=\$(find . -name '*~')

Question 25 Trouver les lignes de l'entrée standard contenant l'un de ces 4 textes : «le chat bleu», «le chat vert», «le chien bleu», «le chien vert»

- grep -E 'le (chien|chat) (vert|bleu)'
 grep -E 'le chat|chien bleu|vert'
 grep -E 'le {chat|chien} {bleu|vert}'
 grep -E 'le [chien|chat] [vert|bleu]'

Question 26 Afficher les noms des entités de tous les sous-répertoires du répertoire courant :

- echo * * echo * echo /* echo * / * echo */*

Question 27 Chercher la chaîne de caractère «\$A» (qui est donc composée des caractères «\$» et «A») dans LES fichiers dont LES chemins sont contenus dans la variable «A»

- grep "\$A" \$A grep "\$A" '\$A' grep '\$A' \$A grep "\$A" "\$A"
 grep \$A '\$A' grep '\$A' "\$A" grep \$A "\$A" grep \$A \$A



Question 28 N'afficher que les noms se terminant par «x» des entités du répertoire courant :

- echo x'*' echo .*x echo "x*" echo x\
 echo \
 echo *x echo '.*x' echo x.* echo *x echo ".*x"

Question 29 Afficher les noms des entités du répertoire courant qui ne commencent pas par une lettre :

- echo ![a-z]* echo [^a,z]* echo [^a-z]* echo ![a-z]*
 echo ![az] echo ![az]* echo ^[a-z]* echo ^[a-z].*

Question 30 Afficher 6 :

- \$(expr 2 \
 expr \$(2 * 3) expr "2*3" expr 2*3 expr (2 * 3)
 expr 2 * 3 expr 2 '*' 3 expr 2"*"3

Question 31 Soit les fichiers contenant un «A» et AUSSI un «B» pas forcément sur la même ligne. Afficher le nom de ces fichiers, on suppose qu'il n'y a pas d'espace dans les noms de fichier. L'option «-l» affiche seulement le chemin du fichier qui contient le texte, pas la ligne elle-même :

- grep -l '(A.*B|B.*A)' * grep -l -e A -e B * grep -l * '(A.*B|B.*A)'
 grep -l A \$(grep -l B *) grep -l A * | grep -l B grep -l A * | grep B

Question 32 Afficher le contenu de tous les fichiers de moins de 1Ko :

- find . -size -1k -exec cat {} \
 find . -size -1k -exec cat {} ; find . -size 1k -exec cat {} \
 find . -exec cat {} \
 find . -exec cat {} \; -size -1k

Question 33 Détruire toutes les entités du répertoire courant en indiquant pour CHACUNE des entités la date de fin de destruction :

- rm * ; date rm * \$(date)
 for V in * ; do rm \$V date ; done for V in * ; do rm * ; echo \$(date) ; done
 rm * date for V in * ; do rm \$V ; date ; done
 for V in * ; do rm "\$V" ; date ; done for V in * ; do rm '\$V' ; echo date ; done

Question 34 Trouver tous les répertoires qui sont dans votre répertoire de connexion :

- find --type=d find .. -name '*' find ~ -type d
 find \~ type=d find / --type "d" find . -type 'd'

Question 35 Afficher la moyenne du nombre contenu dans le fichier «A» et de celui contenu dans le fichier «B» :

- read A<A ; read B<B ; expr (\$A + \$B)/2 echo ((\$(cat A) + \$(cat B)) / 2)
 read A<A ; read B<B ; expr "\$A + \$B)/2" echo expr ((\$A + \$B)/2)
 expr \
 expr '(' \$A + \$B ')' / 2

Question 36 Expression régulière trouvant les lignes contenant un «a» et plus loin un «b» :

- a.*b [a-b] ab* [ab]* a*b a*b* (ab)*

Question 37 Lequel de ces chemins ne mène pas au même endroit que les autres ?

- ../Y ./Y ../Y ../Y/./Y
 ../X/./Y ../Y/ ../Y/ ../Y/



Question 38 Expression régulière étendue trouvant les textes non vides composés d'une suite de paires une_lettre+un_chiffre, par exemple «r2d2» mais pas «c3po» (seulement «c3») :

- ([a-z][0-9])+ [a-z]+[0-9]+ [a-z][0-9]+ [a-z0-9]+ [a-z,0-9]+

Question 39 Remplacer le deuxième mot de chaque ligne par le caractère «=», sans toucher au reste :

- sed -r 's/([!]+) [!]+/\1 =/' sed -r 's/([^]+) [^]+/\1 =/'
 sed -r 's/([!]+) .+/\1 =/' sed -r 's/[^]+ [^]+/\& =/'

Question 40 Mettre dans «G» la concaténation des contenus des fichiers «X/F» et «Y/F» :

- cat X/F >G ; cat Y/F >G ((cd X ; cat F) ; (cd Y ; cat F)) >G
 cd X ; cat F >G ; cd Y ; cat F >>G (cd X ; cat F >G) ; cd Y ; cat F >G

Question 41 Chercher les lignes contenant «i = 0» dans tous les fichiers «.c» et «.h» :

- grep '*.ch' i = 0 grep i = 0 '*.ch'
 grep 'i = 0' *.ch grep 'i = 0' '*.ch'

Question 42 Afficher les noms avec une extension de 3 caractères (comme «.jpg» par exemple) :

- echo .*** echo .??? echo .*??? echo *.??? echo '!'...

Question 43 Comprimer tous les fichiers de plus de 1Mo modifiés il y a plus de 30 jours et qui ne sont pas déjà compressés :

- find . -type f -size +1M -mtime +30 ! -name *.gz | while read A ; do gzip "\$A" ; done
 gzip \$(find . -type f -size >1M -mtime >30 ! -name *.gz)
 find . -type f -size 1M -o -mtime 30 -o ! -name '*.gz' | xargs gzip
 find . \(-type f -size 1M -mtime 30 ! -name '*.gz' \) | gzip
 find . -type f -size +1M -mtime +30 ! -name *.gz -exec gzip {} ;
 find . -type f -size +1M -mtime +30 ! -name '*.gz' -exec gzip {} \;

Question 44 Doubler chacun des mots lus sur l'entrée standard. «la souris triste» devient «la la souris souris triste triste» :

- sed -r 's/[^]+/\1 \1/g' sed -r 's/[!]+/\& \&/g'
 sed -r 's/([^]+)/\1 \1/g' sed 's/[^]*/\& \&/'

Question 45 Enlever le dernier caractère de chacune des lignes lues sur l'entrée standard :

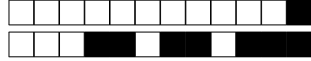
- sed 's/\$.//' sed 's/\. \$.//' sed 's/.\$/\$.//' sed 's/.\$.//'

Question 46 Enlever le dernier mot de chacune des lignes lues sur l'entrée standard :

- sed 's/ .*\$/'' sed 's/.*\$/'' sed 's/[^]*\$/' sed 's/.* .*\/1/''

Question 47 Lister les processus qui contiennent 'main' dans leur nom :

- ps | grep main grep main \$(ps) grep main | ps grep main <ps
 ps main grep main <\$(ps) ps ; main ps main grep



Question 48 Trouver les entités avec l'extension «.c» contenues dans tous les répertoires de la hiérarchie à partir du répertoire courant :

- find . *.c find . -name *.c echo *.c
 find / -name "*.c" find . -name '*.c' echo */*.c

Question 49 Ne pas afficher les erreurs de la commande :

- grep * 2>/dev/null grep * >>null grep * >/dev/null grep * >2
 grep * /dev/null grep * >&2 grep * 2>&1 grep * 1>dev/null

Question 50 Lister les noms des entités directement contenues dans les répertoires «A» et «B» :

- ls A B cat A/* B/* echo A/ B/ cat A B ls A/* B/*