



Systeme Unix

my__select

Contact b-psu-050@epitech.eu



Table des matières

Détails administratifs	2
Sujet	3
Bonus	4
Exemple	5
Fonctions autorisées	6



Détails administratifs

- Les sources doivent être rendues sur le dépôt SVN
- Rendu :
 svn+ssh ://kscm@koala-rendus.epitech.net/myselect-année_promo-login_x
- Votre binaire devra compiler avec un Makefile.
- Votre binaire se nommera `my_select`



Sujet

- Ecrire un programme "my_select" qui prend en paramètre une série d'arguments. La liste d'arguments s'affiche.
- L'utilisateur peut alors se déplacer dans la liste des arguments à l'aide des flèches (la liste est circulaire).
- Un ou plusieurs choix peuvent être sélectionnés ou désélectionnés à l'aide de la touche espace. A chaque sélection effectuée, le curseur doit automatiquement se positionner sur l'élément suivant.
- Dès que l'utilisateur valide la sélection à l'aide de la touche entrée, la liste des choix doit être renvoyée au shell. Les choix renvoyés devront être séparés par le caractère espace. Ceci permettra alors d'utiliser votre programme my_select à l'intérieur d'un script shell (pour faire un "set", par exemple).
- On doit pouvoir écrire les commandes suivantes :

```
1 $> set reponse = 'my_select choix1 choix2 choix3 choix4'
2
3 $> more 'my_select *.c'
4
5 $> rm 'my_select ~/*' NB: N'utilisez cette commande QUE si
6                             vous etes sur que votre my_select
7                             fonctionne. Nous degageons toute
8                             responsabilite en cas de bug de
9                             votre my_select...
10
11 $> ls 'my_select /u/epi1/*' > liste
```

- Vous devrez également gérer le redimensionnement de la fenêtre par l'utilisateur. La liste doit s'afficher sur plusieurs colonnes si la taille de la fenêtre ne comporte pas assez de lignes pour tout afficher en une seule colonne. Si l'utilisateur redimensionne la fenêtre en cours d'utilisation, l'ensemble des choix doit se repositionner automatiquement.
- Si la fenêtre est trop petite pour tout afficher (pas assez ligne et de colonne, alors on affiche un message indiquant que l'utilisateur doit agrandir la fenêtre.
- Si l'utilisateur appuie sur la touche d'échappement, le programme ne doit rien renvoyer au shell.
- Si l'utilisateur appuie sur la touche DELETE ou BACKSPACE, l'élément sur lequel pointe le curseur doit être effacé de la liste. S'il n'y a plus d'élément dans la liste, on quitte.
- choix non sélectionné : texte normal
- choix sélectionné : texte video inversé
- position curseur : texte souligné



Bonus

- Les colonnes défilent de gauche à droite en fonction de la position du curseur quand la fenêtre est trop petite.
- Une belle interface (c'est à l'assistant de juger, pas à vous).
- SuperBonus : Positionnement du curseur lorsque l'on tape une séquence de caractères qui correspond à un élément de la liste (recherche dynamique).



Exemple

```
1 $>./my_select abc toto sguag
2 abc
3 ---
4 toto42
5 sguag
6
7 je tape "sg" :
8
9 abc
10 toto42
11 sguag
12 $>
```



Fonctions autorisées

- ioctl
- tcsetattr
- tcgetattr
- tgetent
- tgoto
- tputs
- tgetstr
- tparm
- open
- close
- write
- malloc
- free
- read
- exit
- signal
- Man utils :
 - termcap
 - termios
 - curs_termcap