

3. Copiez votre programme `shell12.c` dans un nouveau fichier `shell13.c`. Étendez `shell13.c` pour permettre à l'utilisateur de taper des commandes avec un pipe. Par exemple, "`ls /tmp | wc -l`" retourne le nombre de fichiers présents dans le répertoire `/tmp`. Vous aurez besoin d'utiliser la fonction `dup2()`.

Je vous suggère **fortement** de ne pas vous focaliser sur l'interprétation de la ligne de commande. Commencez par une version où le shell demande deux commandes l'une après l'autre puis crée les deux processus et le pipe. Par exemple, pour effectuer la commande "`ls /tmp | wc -l`" vous pouvez réaliser quelque chose comme ceci:

```
> ./shell13
commande1> ls /tmp
commande2> wc -l
15
commande1> exit
>
```

OPTIONNEL: Vous pouvez ensuite étendre votre programme pour supporter la vraie syntaxe "`ls /tmp | wc -l`", voire permettre d'enchaîner davantage de processus: "`sort foo | uniq | wc -l`".

— fin —