Loops

coursera.org/learn/linux-tools-for-developers/supplement/yCeQF/loops

bash permits several looping structures. They are best illustrated by examples.

for

```
done
```

which is equivalent to:

```
$ find . -name "*.o" -exec rm {} ';'
```

or

```
$ find . -name "*.o" | xargs rm
```

showing use of the **xargs** utility.

while

```
#!/bin/sh

ntry_max=4 ; ntry=0 ; password=’ ’

while [[ $ntry -lt $ntry_max ]] ; do

   ntry=$(( $ntry + 1 ))
   echo -n ’Give password:  ’
   read password
   if  [[ $password == "linux" ]] ; then
       echo "Congratulations: You gave the right password on try $ntry!"
       exit 0
   fi
   echo "You failed on try $ntry; try again!"
done

echo "you failed $ntry_max times; giving up"
exit -1
```

until

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

```
ntry_max=4 ; ntry=0 ; password=’ ’

until [[ $ntry -ge $ntry_max ]] ; do

   ntry=$(( $ntry + 1 ))
   echo -n ’Give password:  ’
   read password
   if [[ $password == "linux" ]] ; then
       echo "Congratulations: You gave the right password on try $ntry!"
       exit 0
   fi
   echo "You failed on try $ntry; try again!"

done

echo "you failed $ntry_max times; giving up"
exit -1
```