209 - Tutorial Week 4

Assignment 1

More shell script examples

C programming

Cut

```
#! /bin/sh
# List all the users in /etc/passwd.
FILENAME=/etc/passwd
for user in $(cut -d: -f1 $FILENAME)
do
  echo $user
done
```

More return values

```
adduser() {
  USER=$1; PASSWD=$2
  shift ; shift
  COMMENTS=$@
  useradd -c "${COMMENTS}" $USER
  if [ "$?" -ne "0" ]; then
    echo "Useradd failed”; return 1
  fi
  passwd $USER $PASSWD
  if [ "$?" -ne "0" ]; then
    echo "Setting password failed”; return 2
  fi
  echo "Added user $USER ($COMMENTS) with password
    $PASSWORD”
}
```

Calling the function

```
adduser lionel password TA for 209
if [ "$?" -eq "1" ]; then
  echo "Something went wrong with useradd”
elif [ "$?" -eq "2" ]; then
  echo "Something went wrong with passwd”
else
  echo "User added to the system”
fi
```

Boolean expressions in C

- No boolean type: 0 is false, all other numbers are true.
- 10 < 11 == 1
- 11 < 10 == 0
- i < j < k is equivalent to (i < j) < k.
- Use i < j && j < k

Printf

- man 3 printf: int printf(const char *format, …) - formatted output conversion
- What does that mean?
  - First argument is a string that may contain “conversion specifications”.
  - The number of arguments is variable: 1 + number of conversion specifications.

Conversion specifications

- %d: an integer
- %f: a double
- %.1f: a double, with one digit precision
- %s: a string
- Look at the man page for all the gritty details.

Example

- printf(“i = %d, j = %d\n”, i, j);
- printf(“My float to one decimal place:%.1f\n”, x);

scanf

- int scanf(const char *format, …);
- Dual of printf: input format conversion
- BUT: USES MEMORY ADDRESSES!

- Example: scanf(“%d %d”, &i, &j);