Q2

a)

i. Line 1 declares file as a bash script to be executed. Lines 3-6 are a conditional statement that checks whether there is one argument passed to the script: if not, it echoes correct usage and exits with an error code of 1.

ii. Lines 8-10 set up necessary variables, line 11 runs **which** on smbclient (checks whether the command exists), redirects output to **null** and outputs any errors to the console.

iii. A conditional that checks whether the file with a filename passed as the argument exists. If not, it says the error in the console and exits with an error code of 3.

iv. Line 23 accesses resources from a UofG student's file and prints the filename. Line 25 exits successfully.

b)

Redirects output to **null** in the **dev** directory, then redirects the error ("2>") to stdout ("&1") (console).

c)

Double quotes get the value of any variables present in the string and use those values, while single quotes take only the raw text and print out the variable name, not the value. If, on line 19, single quotes were used, it would literally print out "ERROR: File &{FN_PRINT} does not exist" without interpreting FN_PRINT.

d)

```
#! /bin/bash

if [ $# -eq 1 ]; then
        echo "USAGE: pullprint FILENAME ..."
        exit 1
fi

GUID=gl23x
P_URL=sc-spooler.campus.gla.ac.uk/PullPrint
```

```
which smbclient > /dev/null 2>&1
if [ "$?" -ne "0" ]; then
        echo "ERROR: smbclient not found"
        exit 2
fi

for FN_PRINT in $@
do
        if [ ! -e "$FN_PRINT" ]; then
                echo "ERROR: File ${FN_PRINT} does not exist"
                exit 3
        fi

        smbclient -U campus\\${GUID} //${P_URL} \
                                        -c "print ${FN_PRINT}"
done

exit 0
```

e)

Function:

```
testCommand(){
        which $1 > dev/null 2>&1
        if [ "$?" -ne "0" ]; then
                echo "ERROR: $1 not found"
                exit $2
        fi
}
```

Usage (line 11):

testCommand “smbclient” 2