The purpose of this Lab, is to practice and learn basic shell script commands, per video that we just watched ([LINK](https://drive.google.com/drive/folders/1DkyW-egAQTs_pEmf-fqqmP38Sy1901ks?usp=sharing) / Attached). Please follow instructions outline below. As you execute, please capture the execution results (screen shots) by copy/paste into this document accordingly. Your lab results submission should be your version of this document.

*Note: To view/edit Linux text file on Window OS, please use MS app. “Wordpad” or “Nodepad++”, instead of “NotePad”.*

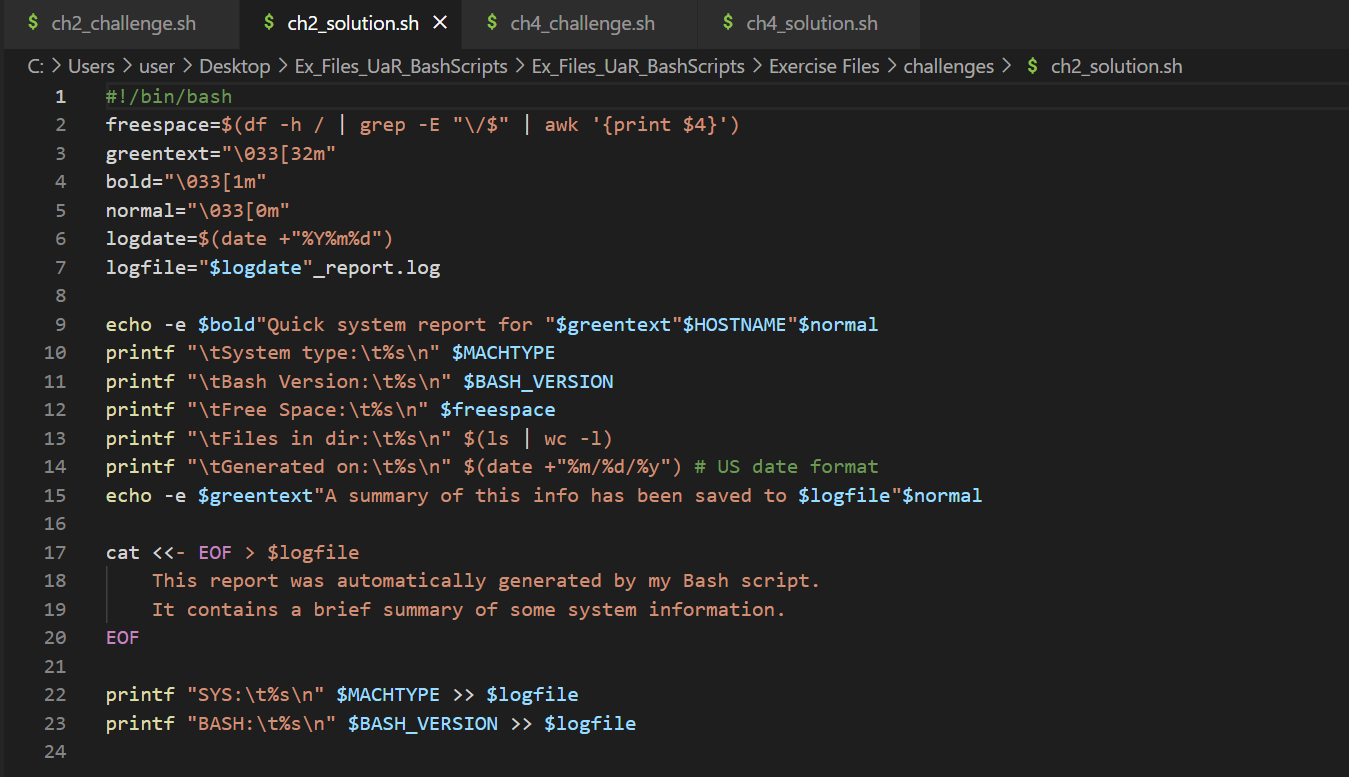
1. Download and extract the exercise file name= “Ex\_Files\_UaR\_BashScripts” from the shell script video that we just watched. (https://drive.google.com/drive/folders/1DkyW-egAQTs\_pEmf-fqqmP38Sy1901ks)



1. Per exercise file above, navigate to its “challenges” sub folder. E.g.



1. Open the relevant .sh file, learn, and make sense of it (you may want to watch the video again).



1. Referring to the solution file: “ch2\_solution.sh”

-----------------------------------------------------------------------------------------

#!/bin/bash

freespace=$(df -h / | grep -E "\/$" | awk '{print $4}')

greentext="\033[32m"

bold="\033[1m"

normal="\033[0m"

logdate=$(date +"%Y%m%d")

logfile="$logdate"\_report.log

echo -e $bold"Quick system report for "$greentext"$HOSTNAME"$normal

printf "\tSystem type:\t%s\n" $MACHTYPE

printf "\tBash Version:\t%s\n" $BASH\_VERSION

printf "\tFree Space:\t%s\n" $freespace

printf "\tFiles in dir:\t%s\n" $(ls | wc -l)

printf "\tGenerated on:\t%s\n" $(date +"%m/%d/%y") # US date format

echo -e $greentext"A summary of this info has been saved to $logfile"$normal

cat <<- EOF > $logfile

This report was automatically generated by my Bash script.

It contains a brief summary of some system information.

EOF

printf "SYS:\t%s\n" $MACHTYPE >> $logfile

printf "BASH:\t%s\n" $BASH\_VERSION >> $logfil

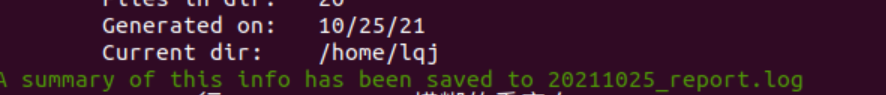
-----------------------------------------------------------------------------------------

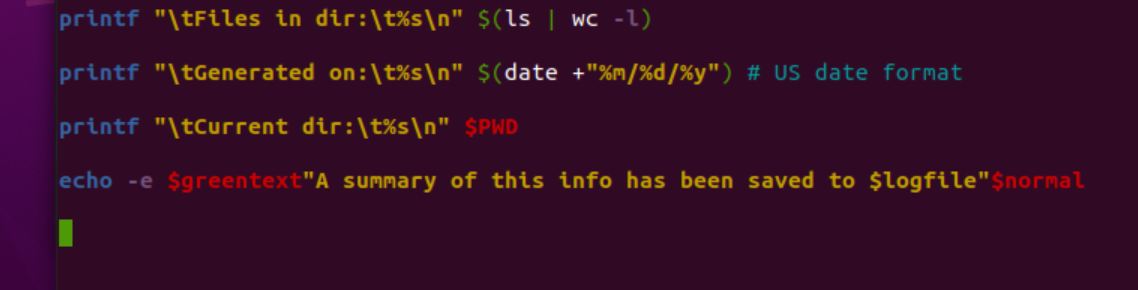
*Make changes to the script file above, so it outputs the following additional information:*

1. *Current working directory*
2. *The system information (“uname” command).*
3. *The script can take 1 input argument (a Linux command, e.g. “whoami”), output the results of executing the command.*

Run/execute your version of the corresponding .sh file, capture the execution result screenshot, and pasted into the document, with your succinct explanation on what it does and what changes that you make.

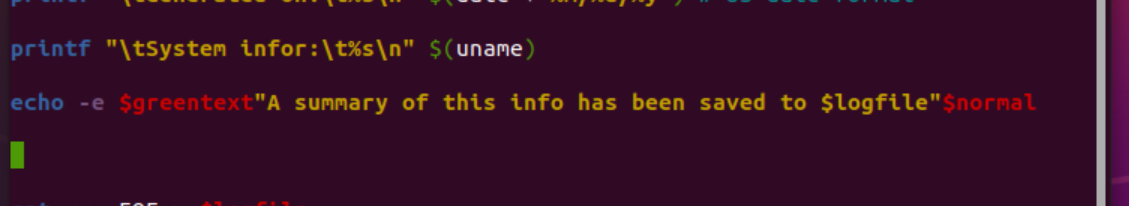
A)

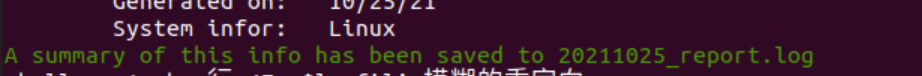




I add a line of code of with PWD as the instruction.

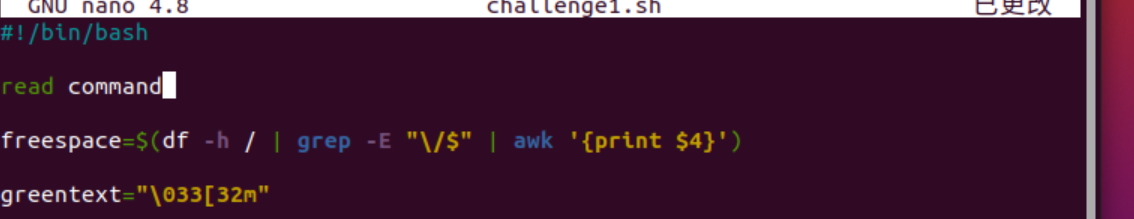
B)

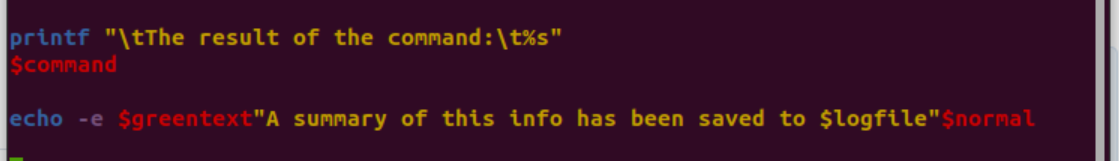


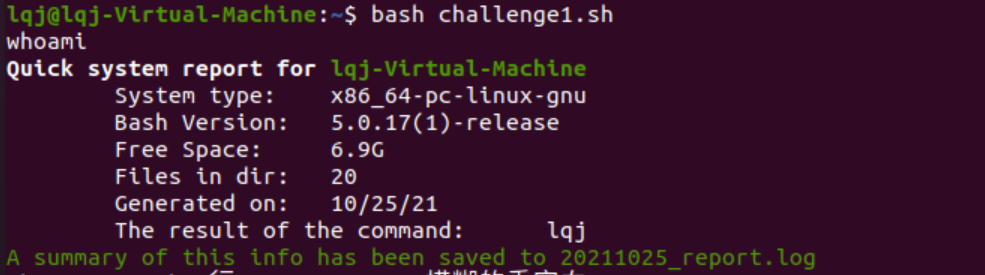


I add a line of code of with uname as the instruction. And use escape character $ as conversion

C)







First, define a command variable command. And use escape character $ as conversion.

4) Referring to the solution file: “ch4\_solution.sh”

-----------------------------------------------------------------------------------------

#!/bin/bash

rand=$RANDOM

secret=${rand:0:1}

function game {

read -p "Guess a random one-digit number! " guess

while [[ $guess != $secret ]]; do

read -p "Nope, try again! " guess

done

echo "Good job, $secret is it! You're great at guessing!"

}

function generate {

echo "A random number is: $rand"

echo -e "Hint: type \033[1m$0 game\033[0m for a fun diversion!"

}

if [[ $1 =~ game|Game|GAME ]]; then

game

else

generate

fi

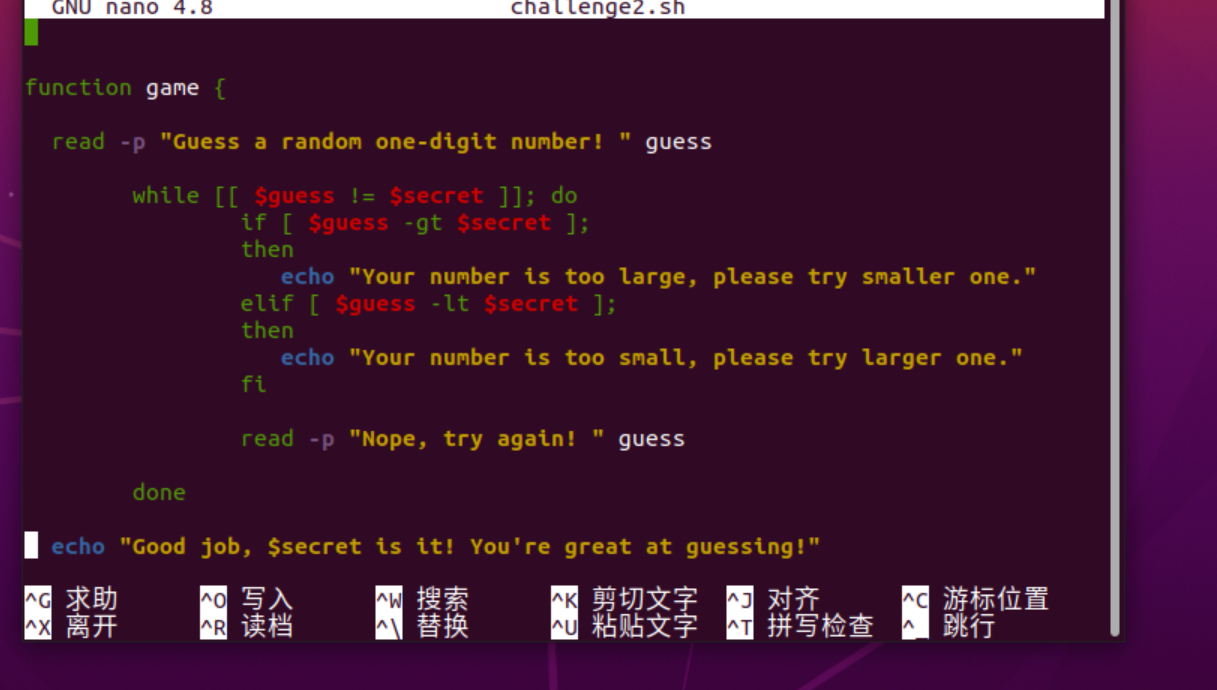
-----------------------------------------------------------------------------------------

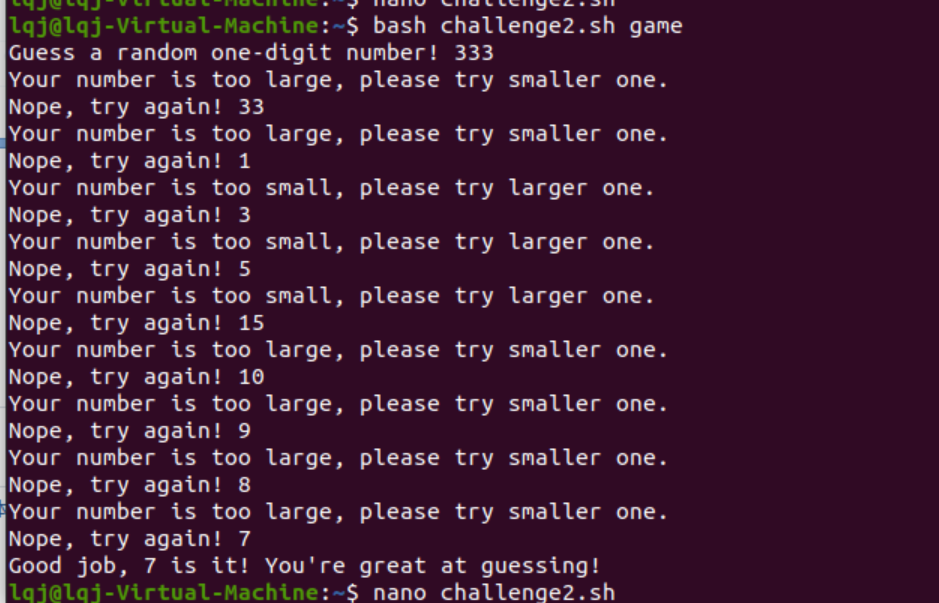
*Add the following additional features to the script.*

1. *If a user’s guess is a miss, inform the user to pick a higher or lower number on the next attempt.*
2. *If a user enters “q | Q | Quit”, exit the script execution, with message “Good Bye”.*

Run/execute your version of the corresponding .sh file, capture the execution result screenshot, and pasted into the document, with your succinct explanation on what it does and what changes that you make.

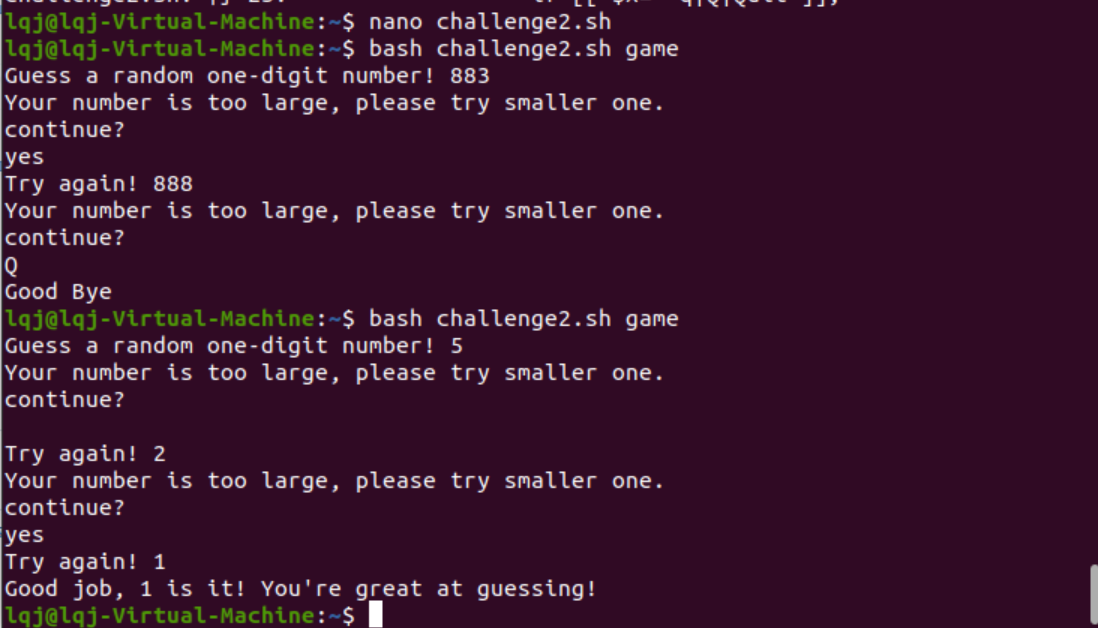
A)

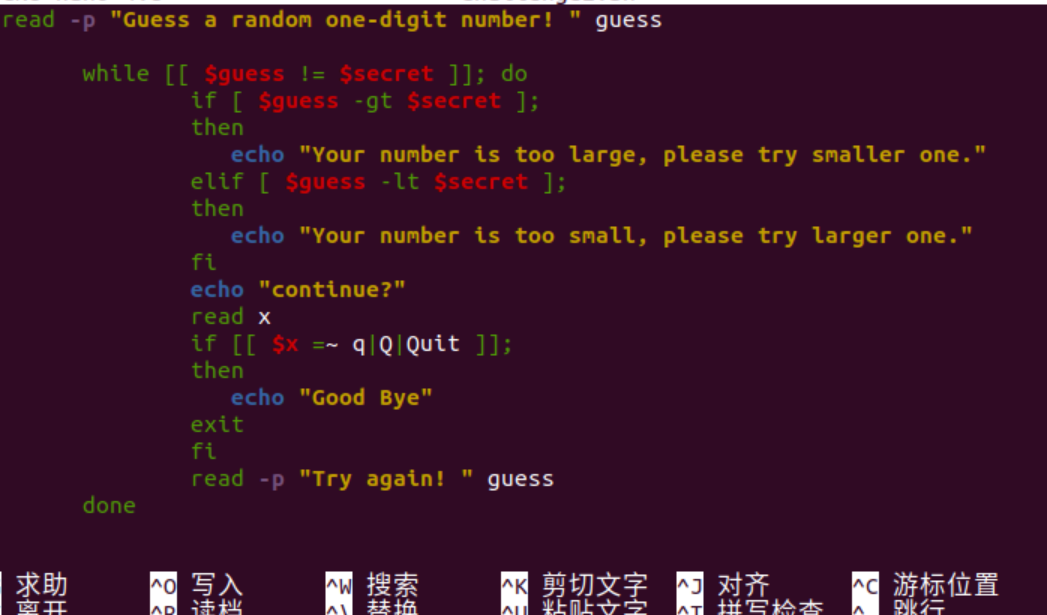




I added an if selection statement to the while loop

B)





I added an ‘if selection’ statement to the while loop, which can choose to exit whether or not.

Notes:

* Your personalized .sh file, its execution should have no syntax error.
* If you are using “Cygwin” to do this Lab, some Linux command may not be pre-installed. You may need to do additional command/package installation (Some command may not be supported at “Cygwin”. However, please do your best to Google search the subject, and see if it can be installed).

**Submission:**

Please submit to online. If that fails, email your results to: [bangpanliang@gmail.com](mailto:jcc4018@qq.com).

The subject of the email should be:

[Your StudentID, Assignment Name]