by Zhivko Stoimchev, 89221056

Because I didn't know what this document should have contained, I just placed the scripts here. I will explain some of the commands I used during creating these scripts.

From task 1:

- **Mkdir** for creating directory
- **Find** to find all chosen photos from the computer, followed by the destination
- **Rm** to remove the folder/directory
- **Tar -zcf** to create compressed directory
- **History** to view all history used in creation of this script

From task 2:

- **Sort** to sort the .csv file
- **Grep** to check if username is located in the file
- Usage of **if statements**

From task 3:

- **Declare -a arr** is for declaring array
- Var1 and var2 are variables
- Using **grep -Eo** to and **-Po** to grep only the text found
- **`${arr[@]}`** `to print the content of the array using for loop`

From task 4:

- **[["CONDITION"]] || "CONDITION"** if condition inside the brackets is fulfilled it skips the statement, if not then it does the thing after ||
- **chmod** to give permission
- creating **alias** in ./.bashrc to run the program from any directory in the terminal
- adding cowsay to **path**
- negation of grep: **!(grep):** what it means it does the opposite if the condition is fulfilled

Task 1

```
mkdir Desktop/photos
find / -name '*jpg' -exec cp "{}" ./Desktop/photos \;
find / -name '*jpeg' -exec cp "{}" ./Desktop/photos \;
find / -name '*png' -exec cp "{}" ./Desktop/photos \;
rm ./Desktop/photos/*.png
tar -zcf ./Desktop/photos_personal.gz ./Desktop/photos
rm -r ./Desktop/photos
history >> zhivko_stoimchev_task1.txt
```

Task 2

```
#!/bin/bash

sort -k4 -n expenses_people.csv -o ordered_expenses.txt
read -p "which username are you interested in: " username
if grep -q $username ./ordered_expenses.txt ; then
      grep $username ordered_expenses.txt | cut -d' ' -f 3,4
else
      echo 'the name cannot be found';
      exit 1
fi

read -p 'are you interested in how many times a certain pattern occurs in
the file? yes/no ' answer
if [ $answer == "yes" ]
then
        read -p 'Which pattern are you interested in? ' pattern
        echo "$pattern is found $(grep -c $pattern ordered_expenses.txt)
times"
else [ $answer == "no" ]
      echo "Goodbye"
      exit
fi
```

Task 3

```
#!/bin/bash

declare -a arr  #declaring an array
read -p 'number of lines in the fragment: ' n

for (( i=0; i<n; i++ ))
do
      read -p 'Enter HTML tag: ' tag
      var1=$(echo $tag | grep -Eo "(http|https)://[a-zA-Z0-9./?=_%:-]*")
      var2=$(echo $tag | grep -Po "(?<=>)([\w\s\.]+)(?=<\/)")
      arr+=("$var1,$var2")
done
for i in "${arr[@]}"; do echo "$i"; done
```

Task 4

```
#!/bin/bash

var=$(git --version)
if( $var == 0 )
then
        [[ -d "cowsay" ]] || git clone
https://github.com/nuwanarti/cowsay.git
      if [[ -d ~/Desktop && -d "cowsay" ]]
      then
            [[ -d ~/Desktop/cowsay ]] || cp -r cowsay ~/Desktop
            chmod 711 ~/Desktop/cowsay/cowsay
            chmod 711 ~/Desktop/cowsay/cowthink
            if !(grep -Fxq "alias cowsay='~/Desktop/cowsay/cowsay'"
      ~/.bashrc)
            then
                  echo "alias cowsay='~/Desktop/cowsay/cowsay'" >>
            ~/.bashrc
                  echo "alias cowthink='~/Desktop/cowsay/cowthink'" >>
            ~/.bashrc
            fi
            PATH=$PATH:~/Desktop/cowsay
            cowsay -f stegosaurus "Thankyou for installing cowsay"
            cowsay –help
            else
                  echo "Desktop/git repo not avaliable"
            fi
else
      echo git not installed
fi
```