EE 463 (Operating Systems)

Section: C3

Semester: Winter 2023

Generalized Bash-Shell Scripts

Name	ID#
Hayan Al-Machnouk	1945954

Course Teacher: Dr. Abdulghani M. Al-Qasimi

Department of Electrical and Computer Engineering

King Abdulaziz University, Jeddah, KSA

Contents

Part A (srm):

Delete files into trash, and restores from trash

Delete trash

```
case "$1" in
"-d")                   # Delete trash Case
    currentTime=$(date +%s) # current time (seconds)
    for file in $HOME/trash/*; do
        if [ ! -e $file ]; then
            echo "the trash is empty"
            break
        fi

        touchTime=$(date -r $file +%s)              # touch time (seconds)
        days=$((($currentTime - $touchTime) / 86400)) # to convert to days, 3600*24=86400
        if [ "$days" -ge 40 ]; then                 # 40 days have passed
            rm -f $file                               # remove file from trash
            grep -wv "$file" $HOME/trash/.srm >L      # remove file from record
            cat L >$HOME/trash/.srm
            rm -f L
        fi
    done
```

```
for name in ${@:2}; do
    if [ ! -e $name ]; then
        echo "$name does not exist"
        continue
    fi

    if echo "$name" | grep "$HOME" >>$HOME/trash/.srm; then
        touch -m -c "$name" # touch time
        mv $name $HOME/trash
    else
        echo "$PWD/$name" >>$HOME/trash/.srm
        touch -m -c "$name" # touch time
        mv $name $HOME/trash
    fi
done
;;
```

Recover trash to source directory

```
"-b") # recover trash to source directory
    if [ "$2" == "file*" ]; then
        for file in $HOME/trash/*; do
            if [ ! -e $file ]; then
                echo "the trash is empty"
                break
            fi
            file=$(echo $file | cut -d '/' -f -5)
            length=2
            sourcePath=$(cat $HOME/trash/.srm | grep "$file")
            movePath=$(echo $sourcePath | cut -d '/' -f -$length)
            while [ -d $movePath ]; do
                movePath=$(echo $sourcePath | cut -d '/' -f -$length)
                length=$(($length + 1))
            done
            movePath=$(echo $sourcePath | cut -d '/' -f -$(($length - 1)))
            mv $HOME/trash/$file $movePath # move ALL trash files to sources

            grep -wv "$movePath" $HOME/trash/.srm >L
            cat L >$HOME/trash/.srm
            rm -f L # remove from record
        done
    else
        for name in ${@:2}; do
            if [ ! -e "$HOME/trash/$name" ]; then
                echo "$name does not exist"
                continue
            fi
            length=2
            sourcePath=$(cat $HOME/trash/.srm | grep "$name")
            movePath=$(echo $sourcePath | cut -d '/' -f -$length)

            while [ -d $movePath ]; do
                movePath=$(echo $sourcePath | cut -d '/' -f -$length)
                length=$(($length + 1))
            done
            movePath=$(echo $sourcePath | cut -d '/' -f -$(($length - 1)))
            mv $HOME/trash/$name $movePath # move trash file to source

            grep -wv "$movePath" $HOME/trash/.srm >L
            cat L >$HOME/trash/.srm # remove from record
            rm L
        done
    fi
    ;;
```

Recover trash to current directory

```
"-m") # recover trash to current directory
    if [ "$2" == "file*" ]; then
        if [ ! -e $file ]; then
            echo "the trash is empty"
            break
        fi
        mv $HOME/trash/* $PWD # move ALL trash files to current directory
    else
        for name in ${@:2}; do
            if [ ! -e "$HOME/trash/$name" ]; then
                echo "$name does not exist"
                continue
            fi

            mv $HOME/trash/$name $PWD # move trash file to current dirctory
        done
    fi
    ;;
```

General case

```
*) # general case
    for file in $*; do
        if [ ! -e "$file" ]; then
            echo "$file does not exist"
            break
        fi

        if echo "$file" | grep "$HOME" >>$HOME/trash/.srm; then
            touch -m -c "$file" # touch time
            mv $file $HOME/trash
        else
            echo "$PWD/$file" >>$HOME/trash/.srm
            touch -m -c "$file" # touch time
            mv $file $HOME/trash
        fi
    done ;;
esac
```

Part B (phone):

Create and maintain phonebook

Adding a record

```
if [ $# -ge 1 ]; then
        case "$1" in
        "-a") # adding a record
                if [ $# -eq 3 ]; then
                        if grep -wq "$2 $3" $HOME/phonelist; then
                                echo "$2 $3 Already Exists"
                                exit 0
                        else
                                echo "$2 $3" >>$HOME/phonelist
                        fi
                else
                        echo "Invalid Arguments, Need a {Name Number} Pair"
                        exit 0
                fi
                ;;
        "-s") # search a record
```

Search a record

```
    "-s") # search a record
            if [ $# -eq 2 ]; then
                    if grep "$2" $HOME/phonelist; then
                            exit 1
                    else
                            echo "$2 Does not Exist"
                            exit 0
                    fi
            else
                    echo "Invalid Arguments, Need only a Name to search"
                    exit 0
            fi
            ;;
    "-r") # replace a record
```

Replace a record

```
"-r") # replace a record
        if [ $# -eq 3 ]; then
                if grep -wv "$2" $HOME/phonelist >val1 &&
                grep -w "$2" $HOME/phonelist >val2; then
                        sed -i "s/$2/$3/g" val2 # stream editor
                        cat val1 >$HOME/phonelist && cat val2 >>$HOME/phonelist
                        rm -f val1 val2
                        exit 1
                else
                        echo "$2 Does not Exist"
                        exit 0
                fi
        else
                echo "Invalid Arguments, Need a {Name Name or Number Number} Pair"
                exit 0
        fi
        ;;
```

Delete a record

```
"-d") # delete records
        if [ $# -eq 3 ]; then
                if grep -wq "$2 $3" $HOME/phonelist; then
                        grep -wv "$2 $3" $HOME/phonelist >val
                        cat val >$HOME/phonelist
                        rm -f val
                        exit 1
                else
                        echo "$2 $3 Does not Exist"
                        exit 0
                fi
        elif [ $# -eq 2 ]; then
                if grep -wq "$2" $HOME/phonelist; then
                        grep -wv "$2" $HOME/phonelist >val
                        cat val >$HOME/phonelist
                        rm -f val
                        exit 1
                else
                        echo "$2 Does not Exist"
                        exit 0
                fi
        else
                echo "Invalid Arguments, Need a {Name Number} Pair or just a Name or Number"
                exit 0
        fi
        ;;
```

Part C (mmerge):

Fills a template message from a list of names

```
if [ ! -e $1 ]; then
        echo "$1 Does not exist"
        exit 0
elif [ ! -e $3 ]; then
        echo "$3 Does not exist"
        exit 0
fi

while read name; do
        echo -e "To: $name \t\t Date: $(date +'%D')" >> "letter-to-$name"
        cat $1 >> "letter-to-$name"
        sed -i "s/$2/$name/g" "letter-to-$name" # stream editor
done < "$3"
```

Template

```
mmerge    template.txt
Part C (mmerge) > template.txt
1
2   NAME The purpose of this letter is to remind you of your outstanding fee
3
4   Best regards, MSHNWQ.
5
```

Unit Tests for all the scripts can be found in the Tests Folder