Faculty of Engineering and Technology

Electrical and Computer Engineering Department

LINUX LABORATORY

(ENCS3130)

Shell Scripting Project Report

Student's Name: Abdalkarim Eiss Student's Number: 1200015

Instructor: Dr. Mohammad Jubran Section: 4

Teaching Assistant: Eng. Ibrahim Injass

Date: 14/5/2022

Abstract

In this project, the system will offer operations related to the student records like display all student records and etc.

Table of Contents

List of Figures

Code

```
# Name: Abdalkarim Nael Eiss          Id: 1200015

# Welcome to the system

printf "\t\t<<<< Welcome to the Student Records System >>>>\n\n"

# To display the set of instrctions to the user

printf ">> Please enter the name of the file containing student records: "

# Declare a variable

filename="$1"


while true                    # Loop to keep the program read the file from the user

do

  read filename       #To read the name of file from the user

  if [ -e "$filename" ]   # To check if the file exist or not

  then

        echo ">>> The file name was read successfully ^_^"                    # To display a message that the file exists to the user

printf "\n1. Show or print student records (all semesters).\n2. Show or print student records for a specific semester."

printf "\n3. Show or print the overall average.\n4. Show or print the average for every semester.\n5. Show or print the total number of passed hours.\n"

printf "6. Show or print the percentage of total passed hours in relation to total F and FA hours.\n7. Show or print the total number of hours taken for every semester.\n"
```

```
printf "8. Show or print the total number of courses taken.\n9. Show or print the total number of labs taken.\n10. Insert the new semester record.\n11. Change in course grade.\n"

#For loop to draw a line between statements

for x in 0 1 2 3

do

printf "....................."

done

# To print a message for the user

printf "\n\n>> Please enter a number from 1 to 11 :  "

char="$1"

read char

#numchars=$(echo -n "$char" | wc -c)

#if [ "$numchars" -ne 1 ]

#then

#echo "Please type a single character"

#exit 1

#fi

case "$char"

in

        #To display student records for all semester

1 ) echo "Student records for all semesters: "

        cat $filename |tr -s '; ' ' ' | tr -s ', ' ' ' | tr -s ' ' ':' | cut -d':' -f1- | tr -s ':' ' ';;
```

```
        #To display student records for a specific semester
2 )     echo "Please enter a number for a specific semester: "
        choose="$2"
        read choose     #read from the user
        if [ "$choose" -eq 1 ]      # if the user choose the first semester
        then
        echo "First semester record: "
        cat $filename |tr -s '; ' ' ' | tr -s ', ' ' ' | tr -s ' ' ':' | tr -s ':' ' ' | grep '/1' | cut -d' ' -f2-
        #To display student records for first semester
        elif [ "$choose" -eq 2 ]
        then
        echo "Second semester record: "
    cat $filename |tr -s '; ' ' ' | tr -s ', ' ' ' | tr -s ' ' ':' | tr -s ':' ' ' | grep '/2' | cut -d' ' -f2-
    elif [ "$choose" -eq 3 ]
        then
        echo "Summer semester record: "
    cat $filename |tr -s '; ' ' ' | tr -s ', ' ' ' | tr -s ' ' ':' | tr -s ':' ' ' | grep '/3' | cut -d' ' -f2-
    else
        echo "Please enter a number of semester from 1 to 3"
        exit 1              #To close from this case if the input not correct
        fi;;
```

3) cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f3 | grep -v "I" | sed 's/FA/50/' | sed 's/F/55/' > Grades.txt #To extract all grades for all semesters and involve handle repeated courses and I courses

sum=`awk '{ sum += $1 } END { print sum }' Grades.txt` #To find the sum of all grades

n=`cat Grades.txt | wc -l` #To find the number of grades

avg=`echo 2 k 0 "$sum" "$n" / + p | dc` #To find the average

#The arithmetic precision is changed with the command k, which sets the number of fractional digits (the number of digits following the point)

#The command p to print out to the screen the top element on the stack

echo "The average for all records is : "$avg"";; #To print and display the average

4) echo "Please enter a number for a specific semester: "

choices="$3"

read choices #To read a number from the user

#For the first semester

if ["$choices" -eq 1]

then

cat $filename | grep "/1" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f3 | grep -v "I" | sed 's/FA/50/' | sed 's/F/55/' > Grades.txt #To extract all grades in the first semester

Total=`awk '{ sum += $1 } END { print sum }' Grades.txt` #To find the sum of grades

Num=`cat Grades.txt | wc -l` #To find the number of grades

AVG=`echo 2 k 0 "$Total" "$Num" / + p | dc` #To find the average

echo "The average for all records in first semester is : "$AVG""

#For the second semester

```
elif [ "$choices" -eq 2 ]
then
cat $filename | grep "/2" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f3 | grep -v "I" | sed 's/FA/50/' | sed 's/F/55/' > Grades.txt       #To extract all records in second semester
Sum=`awk '{ sum += $1 } END { print sum }' Grades.txt`  #To find the sum of grades
num=`cat Grades.txt | wc -l`     #To find the number of grades
Avg=`echo 2 k 0 "$Sum" "$num" / + p | dc` #To find the average
echo "The average for all records in second semester is : "$Avg""
    #For the third semester or the summer semester
    elif [ "$choices" -eq 3 ]
    then
cat $filename | grep "/3" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f3 | grep -v "I" | sed 's/FA/50/' | sed 's/F/55/' > Grades.txt       #To extract all records in the third semester
SUM=`awk '{ sum += $1 } END { print sum }' Grades.txt`  #To find the sum of grades
NUM=`cat Grades.txt | wc -l`     #To find the number of grades
Average=`echo 2 k 0 "$SUM" "$NUM" / + p | dc`      #To find the average
echo "The average for all records in second semester is : "$Average""
else
echo "Please enter a number of semester from 1 to 3"
exit 1
fi;;
5 )        #To extract the number of passed hours for each course
```

```
cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | grep -v "I" | grep -v "FA" | grep -v "F" | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 > passedHours.txt

TotalNumber=`awk '{ sum += $1 } END { print sum }' passedHours.txt`     #To find the total number of passed hours

echo "The total number of passed hours is: "$TotalNumber"";;

6 )     #To extract the number of hours for each course its grade is FA

cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | grep "FA" | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 > FAhours.txt

  #To extract the number of hours for each course its grade is FA

cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | grep -v "FA" | grep "F" | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 > Fhours.txt

if [ -s FAhours.txt ]

then

TotalFAh=`awk '{ sum += $1 } END { print sum }' FAhours.txt`     #To find the number of FA courses hours

  else

TotalFAh="0"

fi

if [ -s Fhours.txt ]

then

TotalFh=`awk '{ sum += $1 } END { print sum }' Fhours.txt`     #To find the number of F courses hours

else

TotalFh="0"
```

```
        fi

        TotalFAF=$(expr "$TotalFAh" + "$TotalFh")

        #To find the number of passed hours

         cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | grep -v "I" | grep -v "FA" | grep -v "F" | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 > copyPassedhours.txt

    if [ -s copyPassedhours.txt ]

    then

    Total=`awk '{ sum += $1 } END { print sum }' copyPassedhours.txt`     #To find the total number of passed hours

    else

        Total="0"

    fi

        # if statement to check the value of total FA and F hours

        if [ "$TotalFAF" -eq 0 ]

        then

        echo "The total of FA and F hours = 0, So the ratio is undefined."

        else

        Ratio=$(expr "$Total" / "$TotalFAF")

        RP=$(("$Ratio" * 100))

        echo "the percentage of total passed hours in relation to total F and FA hours is "$RP"%"

        fi;;
```

```
        #To fined the total of hours for each semester
7 )     echo "Please enter a number for a specific semester: "
    ch="$3"
        read ch
        #For first semester
        if [ "$ch" -eq 1 ]
        then
        #To store all hours in first semester
        cat $filename | grep "/1" | tr -s ' ' ' ' | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 > hs1.txt
        H=`awk '{ sum += $1 } END { print sum }' hs1.txt` #To store the total of all hours in the first semester
        printf "\nThe total hours in first semester = "$H" hours\n\n"
        #For second semester
    elif [ "$ch" -eq 2 ]
    then
    #To store all hours in second semester
    cat $filename | grep "/2" | tr -s ' ' ' ' | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 > hs2.txt
    h=`awk '{ sum += $1 } END { print sum }' hs2.txt`    #To store the total of all hours in the second semester
        printf "\nThe total hours in second semester = "$h" hours\n\n"
    #For third semester
```

```
elif [ "$ch" -eq 3 ]

then

#To store all hours in third semester

cat $filename | grep "/3" | tr -s ' ' ' ' | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 > hs3.txt

hs=`awk '{ sum += $1 } END { print sum }' hs3.txt`          #To store the total of all hours in the third semester

printf "\nThe total hours in third semester = "$hs" hours\n\n"

    else

    printf "\nPlease enter a number between 1 to 3 only !!\n\n"

    exit 1

    fi;;


    #To find the number of all courses

8 )     T=`cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | wc -l` #To sore the number of all courses

    printf "\nTotal number of courses taken = "$T" courses\n\n";;


    #To calculate the number of labs taken

    #To store the number

9 )     N=`cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | cut -d' ' -f2 | perl -pe 's/(.)/\1 /g' | cut -d' ' -f6 | grep "1" | wc -l`

    printf "\nTotal number of labs taken = "$N" labs\n\n";;
```

```
        #To insert a new semester record
10 )    #declare variables
        year="$1"
        code="$1"
        num="$1"
        grade="$1"
        close="$1"
        printf "\nEnter the year and semester as this format (Year/Semester): "
    read year    #To read the year from the user
        printf ""$year"; " >> $filename
  while true
   do
        #Read data from the user
    printf "\nEnter the course code (it should be ENEE or ENCS): "
        read code
        printf "\nEnter the course number (it should be between 2000 to 5999): "
        read num
        printf "\nEnter the grade of course (it should be between 60 to 99 or F or FA or I): "
        read grade
        #To converte the input of string to integer to handle it easily
        if [ "$grade" == "F" ]    # Compare strings
```

```
        then
        s="1"                #a variable to use it in if statement
        grade="55"
    elif [ "$grade" == "FA" ]
    then
    s="1"
    grade="50"
    elif [ "$grade" == "I" ]
    then
    s="1"
    grade="-1"
        else
        s="0"
    fi
        #To check if all inputs meet with conditions
    if [ \( "$code" == "ENEE" -o "$code" == "ENCS" \) -a \( \( "$num" -gt 2000 \) -a \( "$num" -lt 5999 \) \)
-a \( \( \( "$grade" -ge 60 \) -a \( "$grade" -le 99 \) \) -o "$s" -eq 1 \) ]
        then
        #To converte the integer input to character as F, FA and I
        if [ "$grade" -eq 55 ]
        then
        grade="F"
```

```
printf ""$code""$num" "$grade"" >> $filename
    elif [ "$grade" -eq 50 ]
    then
    grade="FA"
printf ""$code""$num" "$grade"" >> $filename
elif [ "$grade" -eq -1 ]
then
    grade="I"
printf ""$code""$num" "$grade"" >> $filename
    else
    printf ""$code""$num" "$grade"" >> $filename
    fi
    printf "\nPLease enter any character to add another course in the same year and semester or (q) to exit: "
    read close          #To read character from the user to complete the insert or not
    #To check the close input
    if [ "$close" == "q" ]        # To check if the input equal q to finish this operation
    then
    printf "\n" >> $filename#To display a new line
    printf "Exit, Added successfully!\n\n"      #To display a message to user
    exit 1     #To finish and close from the loop
    else
```

```
        printf ", " >> $filename  #To display a comma
      continue    #To continue the loop
        fi
        else                #To display an message if inputs did not meet conditions
        printf "\n>>>Try again, one or all of inputs did not meet conditions!!\n"
        continue            #To continue the loop
        fi
    done;;
        # To change a grade for a specific course
11 )    #Declare variables to read data
        course_code="$1"
        course_num="$1"
        NewGrade="$1"
        printf "\nPlease enter the course code (it should be ENEE or ENCS): "
        read course_code          #Read the course code
        printf "\nPlease enetr the course number (it should be between 2000 to 5999): "
        read course_num                    #read the course number
        OldGrade=`cat $filename | grep "EN" | tr -s ' ' ' ' | sort | uniq | cut -d';' -f2 | tr ',' '\n' | grep "$course_code" | grep "$course_num" | cut -d' ' -f3`    # To store the old grade
        printf "\nPlease enter the new grade: (it should be between 60 and 99): "
        read NewGrade             #Read the new grade
```

```
printf "\nThe course code:  "$course_code""$course_num"\nThe old grade:  "$OldGrade"\nThe new grade:  "$NewGrade"\n\n"            #To display the couurse with the old and new grade

printf "\nDo you want to store the new grade instead of the old one? (To confirm this operation enter (y or Y), to cancel enter (n or N):\t"            #To display a message to the user if he want to confirm this change or not

confirm="$1"   #variable to confirm or cancel the operation

read confirm    # read the value from the user

if [ "$confirm" == "n" -o "$confirm" == "N" ]       #To check the input if equal n or N

then

printf "\nCanceled successfully\n\n"      #A message to the user

exit 1

elif [ "$confirm" == "y" -o "$confirm" == "Y" ]      #To check the input if equal y or Y

then

cat $filename | sed "s/"$course_code""$course_num" "$OldGrade"/"$course_code""$course_num" "$NewGrade"/" > temp.txt       #To save the new changes in the temp file

mv temp.txt $filename                # To save the changes into the input file

printf "\n\n\t>>>> Done, changed successfully ^_*\n\n" #Message to the user

else

printf "\nError, please enter (n or N) to cancel or (y or Y) to confirm !!\n\n"

fi;;

* ) echo "PLease enter a number between 1-11 only";;              #To display a message if the entery not correct

esac
```

```
        break                   # To exit and stop the loop and skipping the second condition
    else                                # If the file does not exist
        echo "The file does not exist!!"
        echo "Please enter the name again:  "
        continue                # To back read from the user
    fi
done
```

Testing Cases

When the file is lunched, it asks the user which file they want to read from. There are two cases, the first case if the user entered the name of file correctly and the second case if the user entered it wrongly as shown in the figures below.

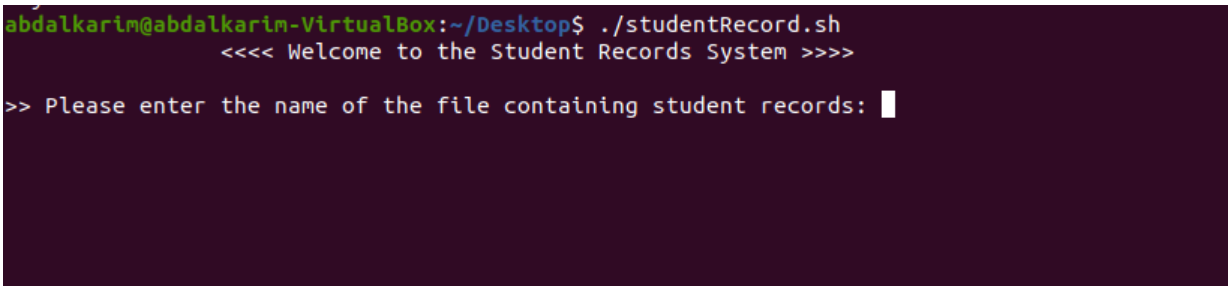

Figure 1: System Interface

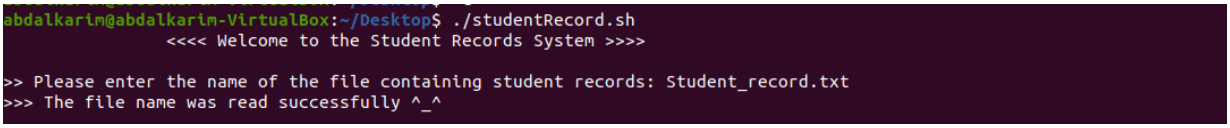

Figure 2: Correct Entry Case

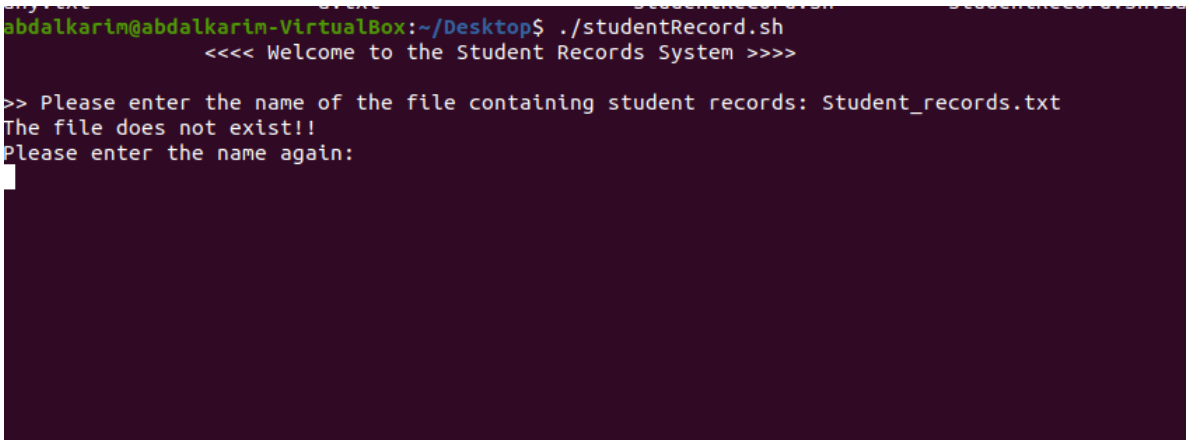

Figure 3: Wrong Entry Case

I applied operations to the Student_record.txt file as shown in figures below:

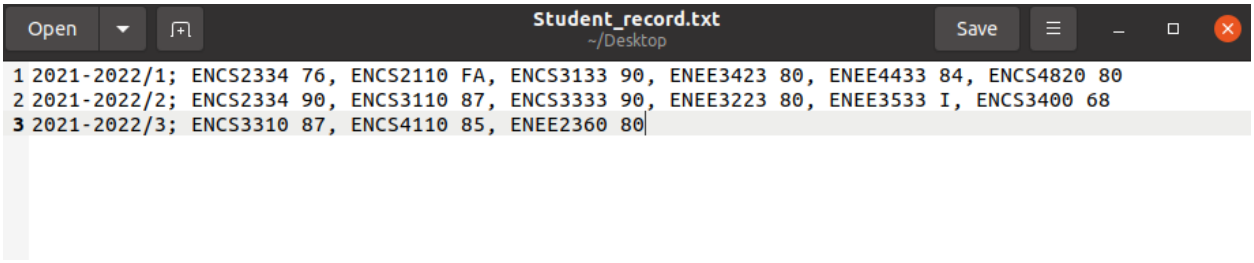

Figure 4: Data of The File

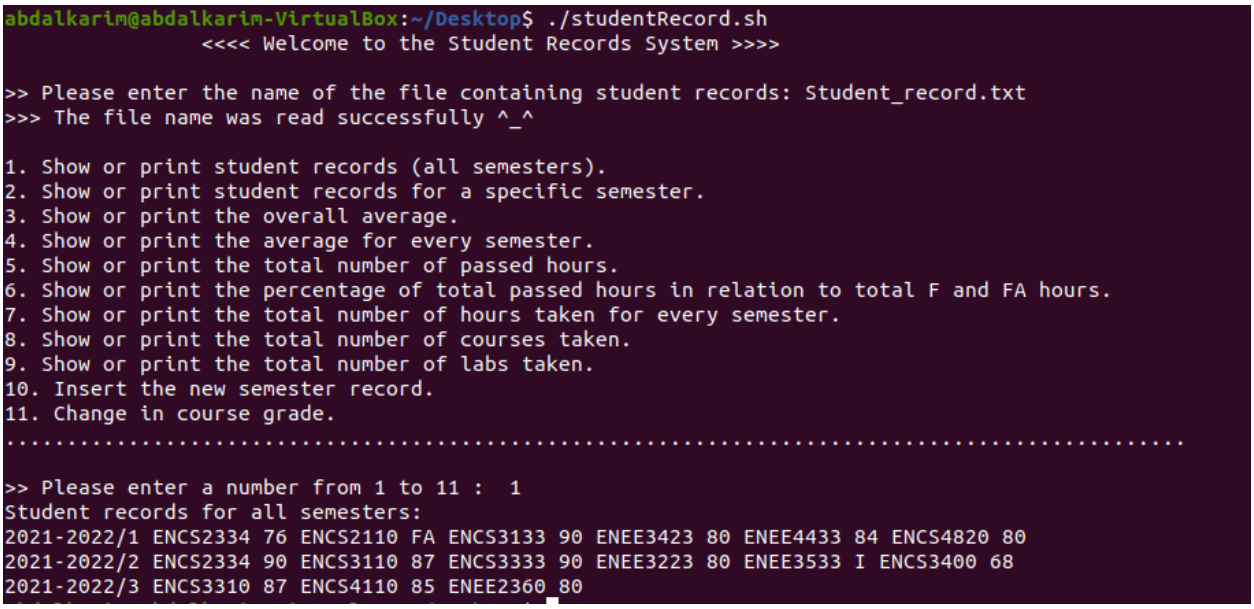

Figure 5: First Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
               <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
................................................................................................

>> Please enter a number from 1 to 11 :  2
Please enter a number for a specific semester:
1
First semester record:
ENCS2334 76 ENCS2110 FA ENCS3133 90 ENEE3423 80 ENEE4433 84 ENCS4820 80
```

Figure 6: First Case in Second Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
               <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
................................................................................................

>> Please enter a number from 1 to 11 :  2
Please enter a number for a specific semester:
2
Second semester record:
ENCS2334 90 ENCS3110 87 ENCS3333 90 ENEE3223 80 ENEE3533 I ENCS3400 68
```

Figure 7: Second Case in Second Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
               <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
.....................................................................................

>> Please enter a number from 1 to 11 :  2
Please enter a number for a specific semester:
3
Summer semester record:
ENCS3310 87 ENCS4110 85 ENEE2360 80
```

Figure 8: Third Case in Second Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
               <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
.....................................................................................

>> Please enter a number from 1 to 11 :  3
The average for all records is : 80.50
```

Figure 9: Third Operation

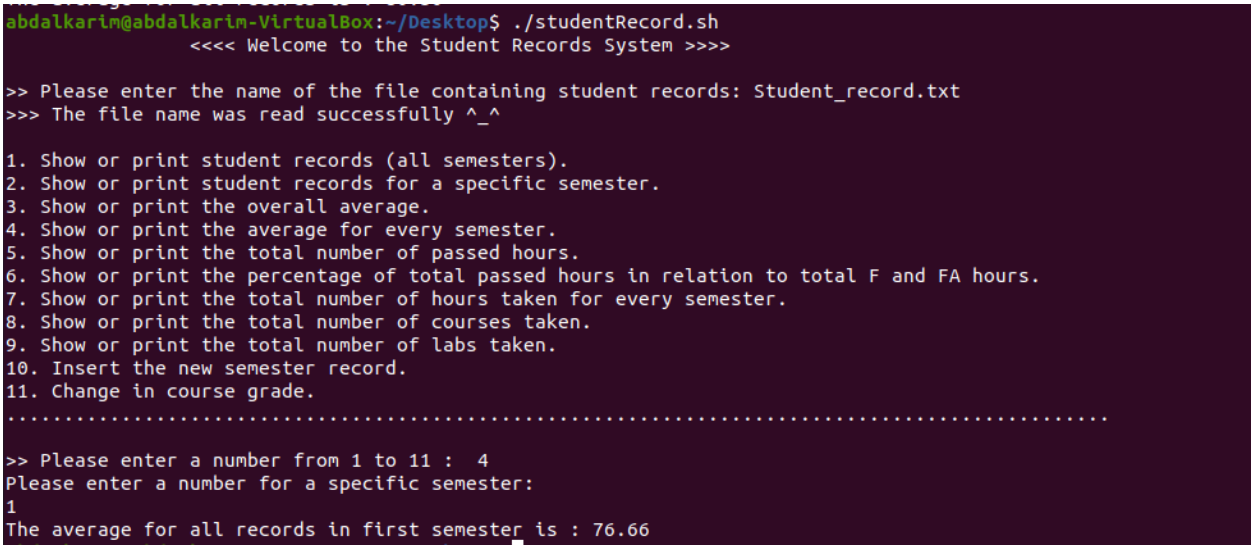

Figure 10: First Case in Fourth Operation

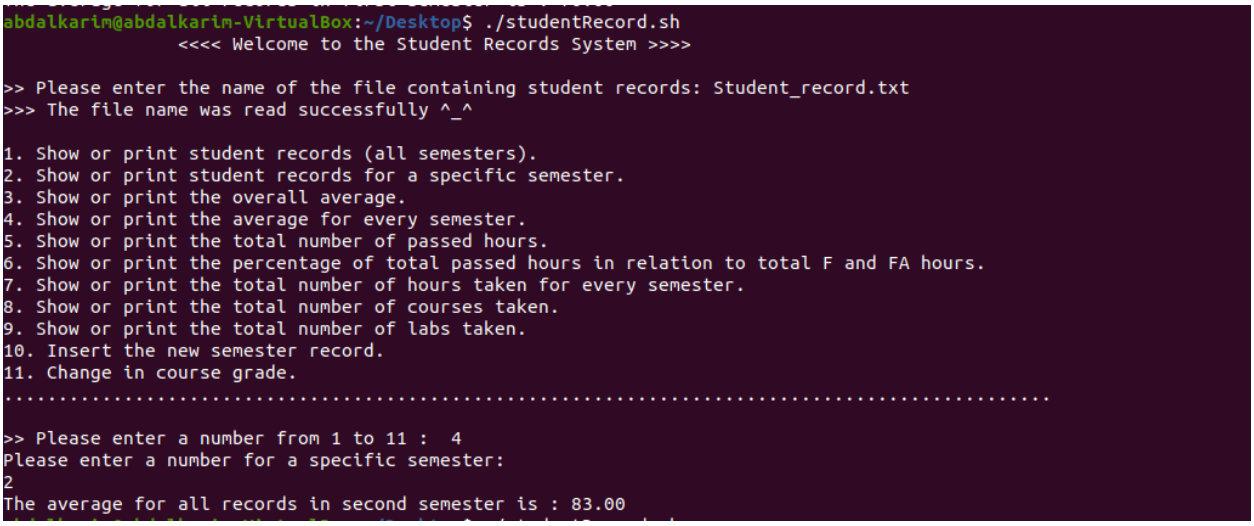

Figure 11: Second Case in Fourth Operation

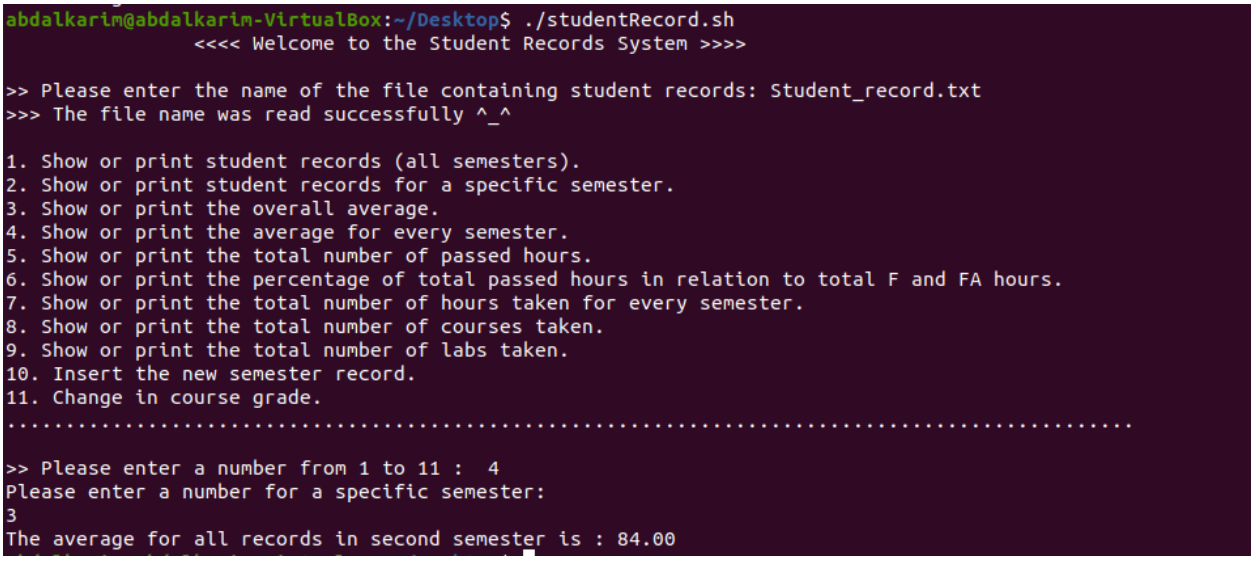

Figure 12: Third Case in Fourth Operation

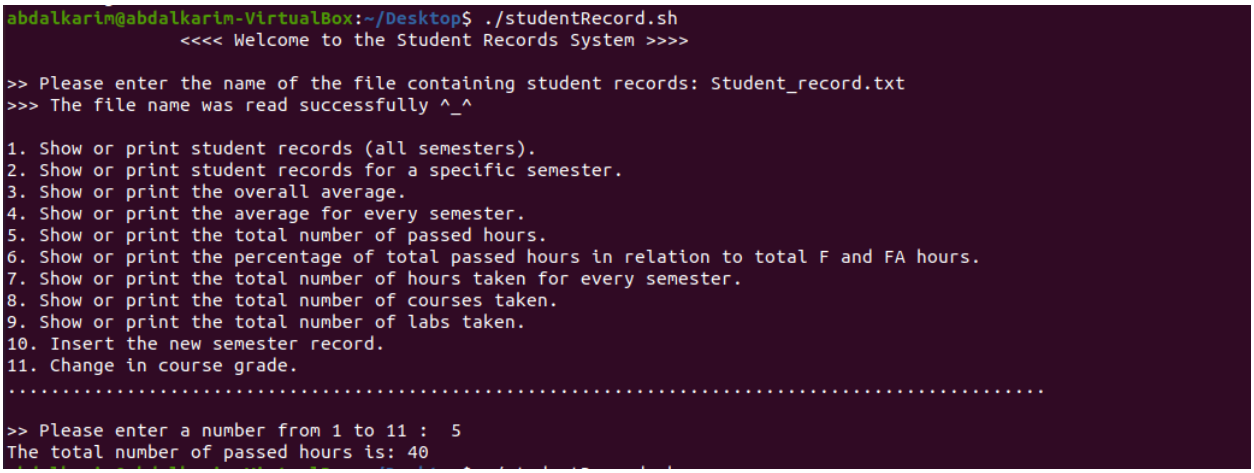

Figure 13: Fifth Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
                <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
................................................................................

>> Please enter a number from 1 to 11 :  6
the percentage of total passed hours in relation to total F and FA hours is 4000%
```

Figure 14: Sixth Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
                <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
................................................................................

>> Please enter a number from 1 to 11 :  7
Please enter a number for a specific semester:
1

The total hours in first semester = 21 hours
```

Figure 15: First Case in Seventh Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
               <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
................................................................................................

>> Please enter a number from 1 to 11 :  7
Please enter a number for a specific semester:
2

The total hours in second semester = 18 hours
```

Figure 16: Second Case in Seventh Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
                <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
................................................................................................

>> Please enter a number from 1 to 11 :  7
Please enter a number for a specific semester:
3

The total hours in third semester = 7 hours
```

Figure 17: T*hird Case in Seventh Operation*

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
                <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
.....................................................................................

>> Please enter a number from 1 to 11 :  8

Total number of courses taken = 15 courses
```

Figure 18: Eighth Operation

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
                <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
.....................................................................................

>> Please enter a number from 1 to 11 :  9

Total number of labs taken = 4 labs
```

Figure 19: Ninth Operation

- In 10th Operation there are two cases (Right and Wrong Cases)

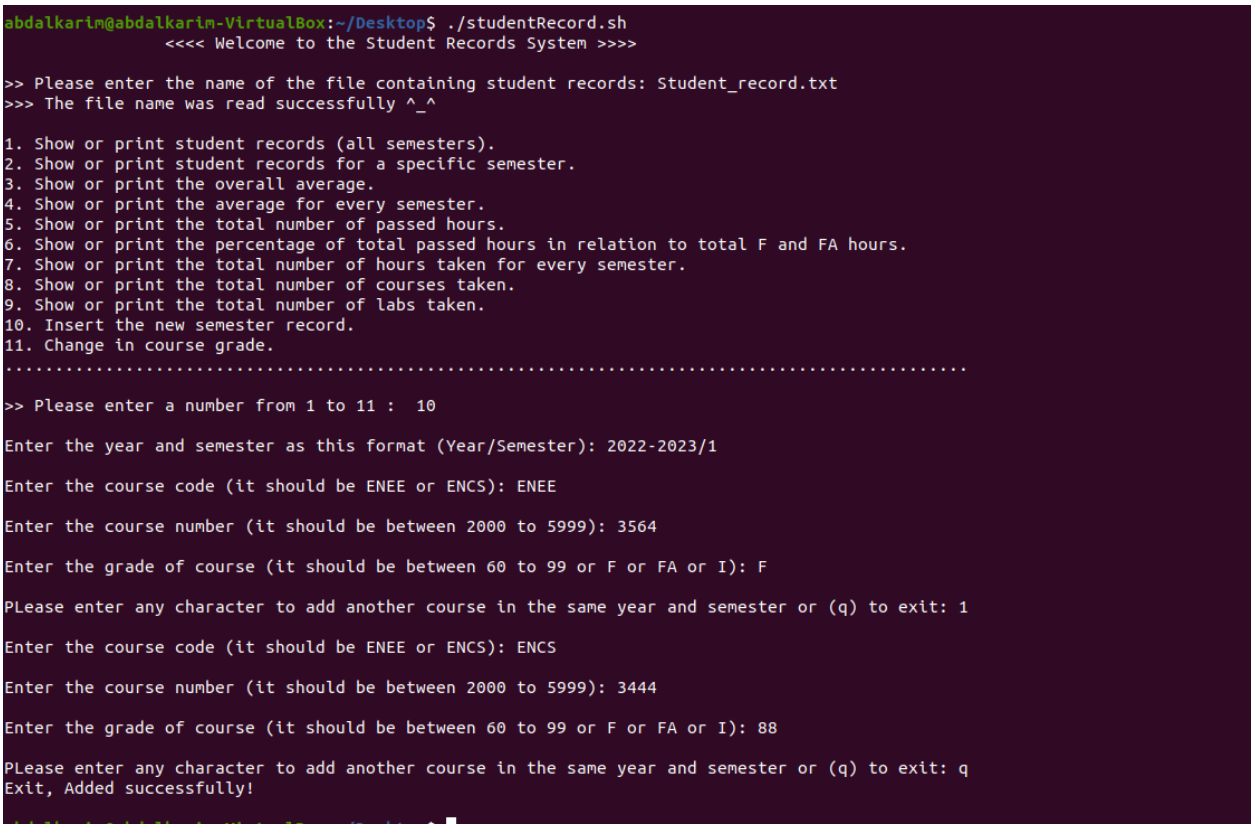

Figure 20: Right Case in Tenth Operation

- **In each addition process, the program will show a message to the user to choose q to exit or any character to complete the insert process.**

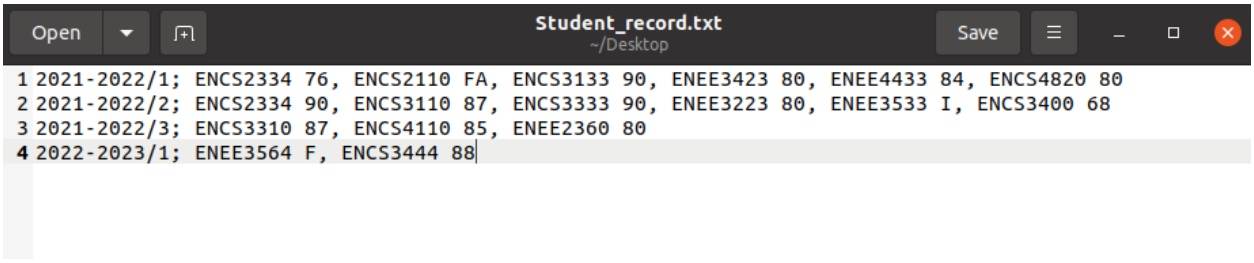

Figure 21: Insert to The File

```
abdalkarim@abdalkarim-VirtualBox:~/Desktop$ ./studentRecord.sh
                <<<< Welcome to the Student Records System >>>>

>> Please enter the name of the file containing student records: Student_record.txt
>>> The file name was read successfully ^_^

1. Show or print student records (all semesters).
2. Show or print student records for a specific semester.
3. Show or print the overall average.
4. Show or print the average for every semester.
5. Show or print the total number of passed hours.
6. Show or print the percentage of total passed hours in relation to total F and FA hours.
7. Show or print the total number of hours taken for every semester.
8. Show or print the total number of courses taken.
9. Show or print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
................................................................................................

>> Please enter a number from 1 to 11 :  10

Enter the year and semester as this format (Year/Semester): 2022-2023/2

Enter the course code (it should be ENEE or ENCS): MATH

Enter the course number (it should be between 2000 to 5999): 1411

Enter the grade of course (it should be between 60 to 99 or F or FA or I): 99

>>>Try again, one or all of inputs did not meet conditions!!

Enter the course code (it should be ENEE or ENCS):
```

Figure 22: Wrong Case in Tenth Operation

- **Wrong Case occurs when one or more one inputs did not meet conditions.**

- **In 11th operation, there is two cases, the first case is displaying the old and new grade with change it in the input file, the second case is displaying the old and new grade without change in the file.**

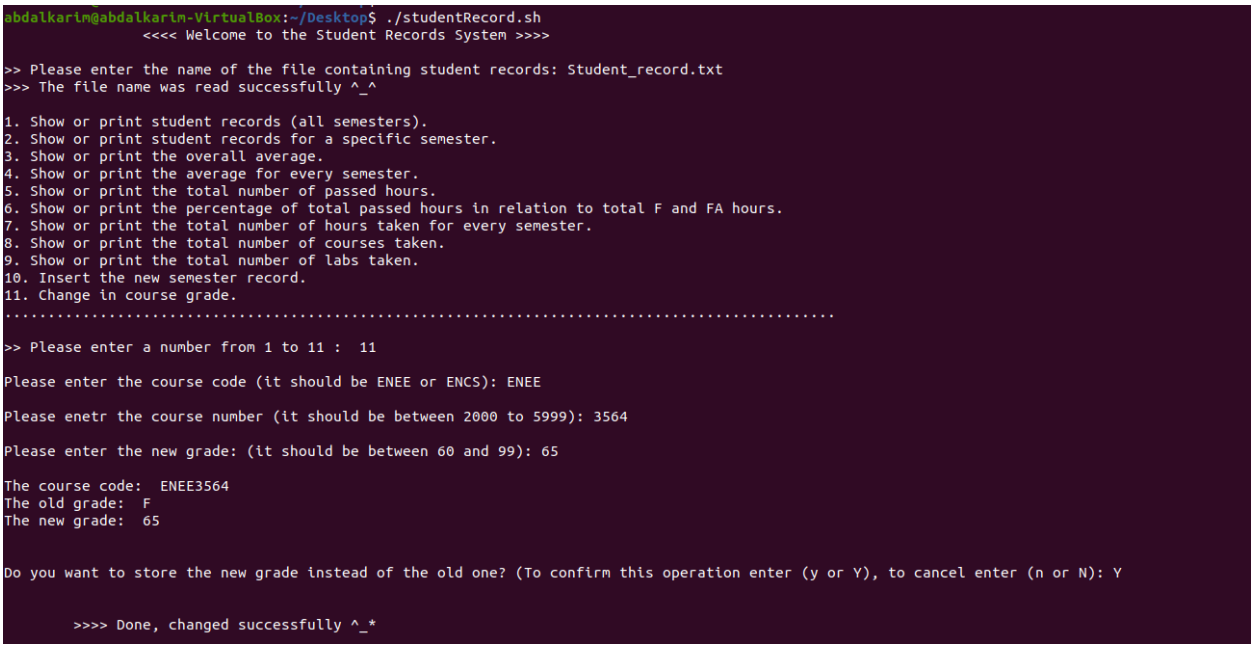

Figure 23: Case One in the 11th Operation

- Changed the grade of ENEE3564 from F to 65 as shown below:

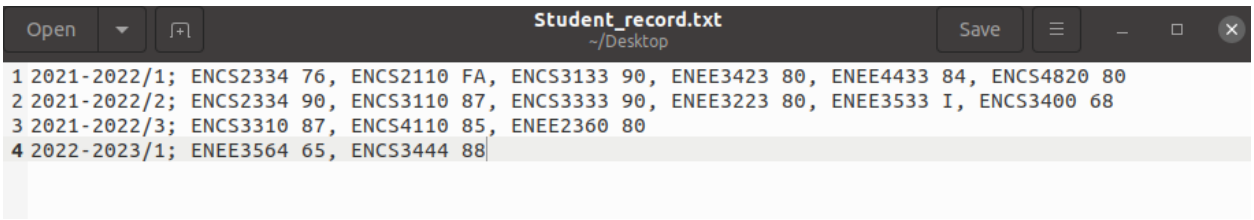

Figure 24: After Changed

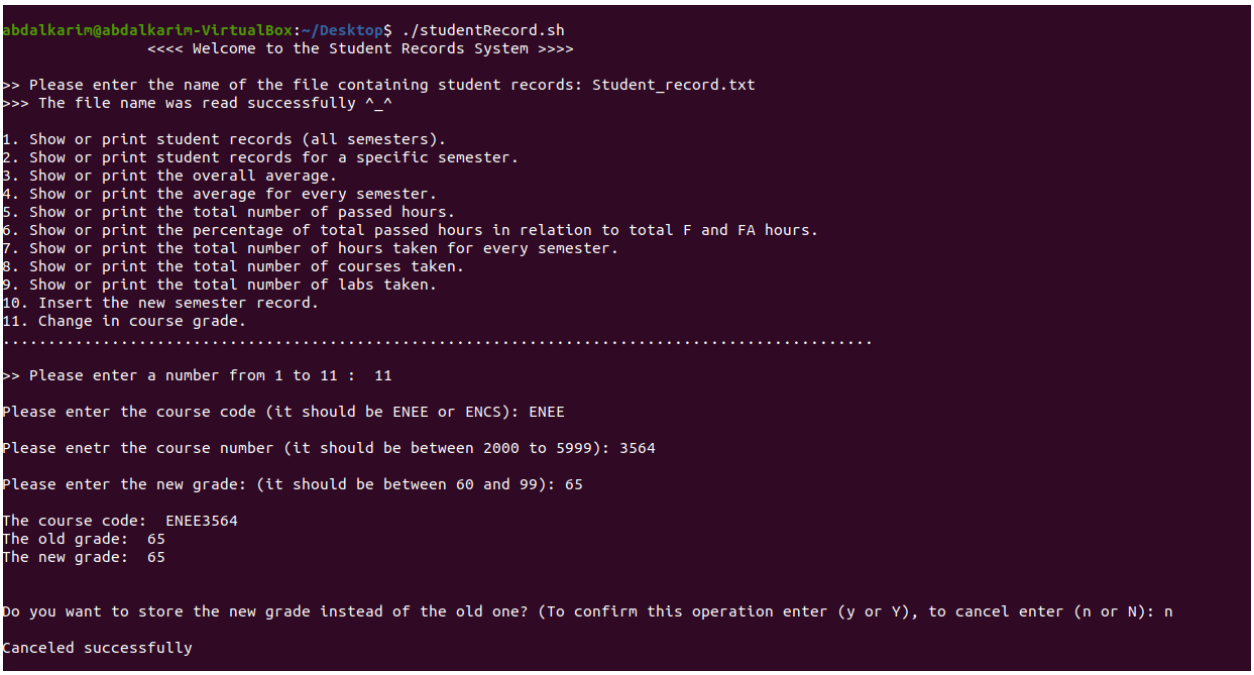

Figure 25: Case Two in the 11th Operation