



Faculty of Engineering & Technology
Electrical & Computer Engineering Department

ENCS3130

Prepared by:

Nidal Zabade 1200153

Partner name:

Momen Assaf 1191529

Instructor: Dr. Aziz Qaroush

Assistant: Eng. Shahd Muhareb

Date: 8/5/2022

Code Idea:-

The code used to do some operation on Student record from text file. First operation is to print all the semesters for the student. The second operation is to print specific semester based on the semester number given. The third operation is to compute the overall average for all semesters. The fourth operation is to print the average for each semester. The fifth operation is to print the total passed hours for the student. The sixth operation is to print the percentage of total passed hours in relation to total F and FA hours. The Seventh operation is to print the total number of hours taken for every semester. The eighth operation is to Show or print the total number of courses taken. The ninth operation is Show or print the total number of labs taken. The tenth operation is to insert the new semester record. The last operation is to change in course grade

The Code:-

```

1 #!/bin/sh
2
3
4 #print_specific_semester() function used to Show or print student records for a specific semester by giving the semester number
5
6 print_specific_semester(){
7 while true
8 do
9 echo "Enter the semester"
10 read semester #read the semester number
11 temp2=$( echo "$semester" | cut -c 11 ) #read if the semester is valid
12 if [ "$temp2" -eq 1 -o "$temp2" -eq 2 -o "$temp2" -eq 3 ] #make sure if the semester 1 represents the first semester (2 for the second semester, 3 for the
13 #summer semester).
14 then
15 break
16 else
17 echo "\nThere is no such semester\n"
18 continue
19 fi
20 done
21 while read line; do #while loop to read line in the file
22 temp=$( echo "$line" | cut -f1 -d";" )
23 if [ "$temp" = "$semester" ]
24 then
25 echo "\n$line\n"
26 return
27 fi
28 done < "$filename"
29 echo "\nThere is no such semester\n"
30 return
31 }
32
33
34
35 #print_overall_avg() function used to Show or print the overall average
36 print_overall_avg(){
37 echo "\n"
38 #variables
39 SUM=0
40 HOURS=0
41 AVG=0
42 #creating files for the operation
43 touch marks
44 touch hours
45 > marks
46 > hours
47 > marks_hours
48
49 > marks_hours
50 while read line; do #read the file
51 temp=$( echo "$line" | sed -n -e 's/^\.; //p' ) #cut the lines just to show the courses and grades
52 echo "$temp"
53 for val in $temp; do
54 val=$( echo "$val" | cut -f1 -d";" )
55 l=$(echo "$val" | grep [^0-9] | wc -c) #make sure that all grades are valid
56 if [ "$l" -ge 0 -a "$l" -le 3 ]
57 then
58 if [ "$val" = "FA" ] #FA (Fail Absent counted as 50).
59 then
60 val=50
61 elif [ "$val" = "F" ] #F (Fail and counted as 55)
62 then
63 val=55
64 elif [ "$val" = "I" ] #I (incomplete)
65 then
66 val=0
67 fi
68 echo "$val" >> marks
69 else
70 echo "$val" | cut -c 6 >> hours
71 fi
72 done
73 done < "$filename"
74 touch marks_hours
75 paste marks hours > marks_hours
76 while read line2; do
77 temp1=$( echo "$line2" | cut -f1 )
78 temp2=$( echo "$line2" | cut -f2 )
79 if [ "$temp1" -eq 0 ]
80 then
81 continue
82 else
83 SUM=$((SUM+((temp1*temp2)))) #Total number of grades by hours
84 HOURS=$((HOURS+temp2)) #Total number of hours
85 fi
86 done < marks_hours
87 AVG=$((SUM/HOURS)) #compute the average
88 echo "\n\nThe Total Points: $SUM\n"
89 echo "The Total Hours: $HOURS\n"
90 echo "-----"
91 echo "|
92 echo "|
93 echo "| The overall average: \"$AVG\"
94 echo "|
95 echo "-----"
96

```

```

93 echo "\n"
94 }
95
96
97 #print_semester_avg() function used for Show or print the average for every semester
98 print_semester_avg(){
99 #creating files for the operation
100 touch marks
101 touch hours
102 > marks
103 > hours
104 > marks_hours
105 while read line; do #read the file
106 temp=$( echo "$line" | sed -n -e 's/^.*; //p' ) #cut the lines just to show the courses and gardes
107 echo "$temp"
108 for val in $temp; do
109 val=$( echo "$val" | cut -f1 -d",")
110 l=$(echo "$val" | grep [^0-9] | wc -c) #make sure that all grades are valid
111 if [ "$l" -ge 0 -a "$l" -le 3 ]
112 then
113 if [ "$val" = "FA" ] #FA (Fail Absent counted as 50).
114 then
115 val=50
116 elif [ "$val" = "F" ] #F (Fail and counted as 55)
117 then
118 val=55
119 elif [ "$val" = "I" ] #I (incomplete)
120 then
121 val=0
122 fi
123 echo "$val" >> marks
124 else
125 echo "$val" | cut -c 6 >> hours
126 fi
127 done
128 touch marks_hours
129 paste marks hours > marks_hours
130 while read line2; do #loop to compute the average line by line aka semester by semester
131 temp1=$( echo "$line2" | cut -f1 )
132 temp2=$( echo "$line2" | cut -f2 )
133 if [ "$temp1" -eq 0 ]
134 then
135 continue
136 else
137 SUM=$((SUM+((($temp1*$temp2)))) #Total number of grades by hours per semester

```

```

137 SUM=$((SUM+${temp1*temp2})) #Total number of grades by hours per semester
138 HOURS=$((HOURS+temp2)) #Total number of hours per semester
139 fi
140 done < marks_hours
141 > marks
142 > hours
143 > marks_hours
144 AVG=$((SUM/HOURS)) #compute the average per semester
145 echo "\nTotal Points: $SUM\n"
146 echo "Total Hours: $HOURS\n"
147 echo "-----"
148 echo "|                                     |"
149 echo "|               The overall average: "$AVG"               |"
150 echo "|                                     |"
151 echo "-----"
152 echo "\n"
153 SUM=0
154 HOURS=0
155 done < "$filename"
156 }
157
158
159 #total_passed_hours() function used for Show or print the total number of passed hours
160 total_passed_hours(){
161 echo "\n"
162 HOURS=0
163 #creating files for the operation
164 touch marks
165 touch hours
166 > marks
167 > hours
168 > marks_hours
169 while read line; do #read the file
170 temp=$( echo "$line" | sed -n -e 's/^.*; //p' ) #cut the lines just to show the courses and grades
171 echo "$temp"
172 for val in $temp; do
173 val=$( echo "$val" | cut -f1 -d"," )
174 l=$(echo "$val" | grep [^0-9] | wc -c) #make sure that all grades are valid
175 if [ "$l" -ge 0 -a "$l" -le 3 ]
176 then
177 if [ "$val" = "FA" ] #FA (Fail Absent counted as 50).
178 then
179 val=50
180 elif [ "$val" = "F" ] #F (Fail and counted as 55)
181 then
182 val=55
183 elif [ "$val" = "I" ] #I (incomplete)
184 then
185 val=0
186 fi
187 echo "$val" >> marks
188 else
189 echo "$val" | cut -c 6 >> hours
190 fi
191 done
192 done < "$filename"
193 touch marks_hours
194 paste marks hours > marks_hours
195 while read line2; do #loop to compute the Total Passed Hours line by line aka semester by semester
196 temp1=$( echo "$line2" | cut -f1 )
197 temp2=$( echo "$line2" | cut -f2 )
198 if [ "$temp1" -eq 0 -o "$temp1" -eq 55 -o "$temp1" -eq 50 ]
199 then
200 continue
201 else
202 HOURS=$((HOURS+temp2)) #Total number of hours per semester
203 fi
204 done < marks_hours
205 echo "-----"
206 echo "|                                     |"
207 echo "|               Total Passed Hours: "$HOURS"               |"
208 echo "|                                     |"
209 echo "-----"
210 }
211
212
213 #percentage_of_passed_hours() function used to Show or print the percentage of total passed hours in relation to total F and FA hours
214 percentage_of_passed_hours(){
215 echo "\n"
216 #variables
217 HOURS_F=0
218 HOURS_FA=0
219 PRE=0
220 #creating files for the operation
221 touch marks
222 touch hours
223 > marks
224 > hours
225 > marks_hours
226 while read line; do #read the file
227 temp=$( echo "$line" | sed -n -e 's/^.*; //p' ) #cut the lines just to show the courses and grades
228 for val in $temp; do

```

```

229 val=$( echo "$val" | cut -f1 -d",")
230 l=$(echo "$val" | grep [^0-9] | wc -c) #make sure that all grades are valid
231 if [ "$l" -ge 0 -a "$l" -le 3 ]
232 then
233 if [ "$val" = "FA" ] #FA (Fail Absent counted as 50).
234 then
235 val=50
236 elif [ "$val" = "F" ] #F (Fail and counted as 55)
237 then
238 val=55
239 elif [ "$val" = "I" ] #I (incomplete)
240 then
241 val=0
242 fi
243 echo "$val" >> marks
244 else
245 echo "$val" | cut -c 6 >> hours
246 fi
247 done
248 done < "$filename"
249 touch marks_hours
250 paste marks_hours > marks_hours
251 while read line2; do
252 temp1=$( echo "$line2" | cut -f1 )
253 temp2=$( echo "$line2" | cut -f2 )
254 if [ "$temp1" -eq 55 -o "$temp1" -eq 50 ]
255 then
256 HOURSF=$((HOURSF+$temp2)) #compute Total Failed Hours
257 elif [ "$temp1" -eq 0 ]
258 then
259 continue
260 else
261 HOURSPP=$((HOURSPP+$temp2)) #compute Total PASSED Hours
262 fi
263 done < marks_hours
264 echo "Failed Hours: "$HOURSF"\n"
265 echo "Passed Hours: "$HOURSPP"\n"
266 PERF=$((($((HOURSF*100))/HOURSPP)) #compute Percentage of Failed Hours
267 echo "-----"
268 echo "| |"
269 echo "| Percentage of Failed Hours: "$PERF"% |"
270 echo "| |"
271 echo "-----"
272

```

```

273 PERP=$((100-(((($HOURS*100))/HOURS)))) #compute Percentage of PASSED Hours
274 echo "-----"
275 echo "|                                     |"
276 echo "|          Percentage of PASSED Hours: "$PERP"%          |"
277 echo "|                                     |"
278 echo "-----"
279 echo "\n"
280 }
281
282
283 #hours_taken() function used to Show or print the total number of hours taken for every semester.
284 hours_taken(){
285 #creating files for the operation
286 touch marks
287 touch hours
288 > marks
289 > hours
290 > marks_hours
291 while read line; do #read the file
292 temp=$( echo "$line" | sed -n -e 's/^.*; //p' ) #cut the lines just to show the courses and gardes
293 echo "\n$temp"
294 for val in $temp; do
295 val=$( echo "$val" | cut -f1 -d", " )
296 l=$(echo "$val" | grep [^0-9] | wc -c) #make sure that all grades are valid
297 if [ "$l" -ge 0 -a "$l" -le 3 ]
298 then
299 if [ "$val" = "FA" ] #FA (Fail Absent counted as 50).
300 then
301 val=50
302 elif [ "$val" = "F" ] #F (Fail and counted as 55)
303 then
304 val=55
305 elif [ "$val" = "I" ] #I (incomplete)
306 then
307 val=0
308 fi
309 echo "$val" >> marks
310 else
311 echo "$val" | cut -c 6 >> hours
312 fi
313 done
314 touch marks_hours
315 paste marks_hours > marks_hours
316 while read line2; do
317 temp1=$( echo "$line2" | cut -f1 )
318 temp2=$( echo "$line2" | cut -f2 )

```

```

319 if [ "$temp1" -eq 0 ]
320 then
321 continue
322 else
323 HOURS=$((HOURS+$temp2)) # compute total hours taken
324 fi
325 done < marks_hours
326 > marks
327 > hours
328 > marks_hours
329 echo "\n[Number of Hours Taken: "$HOURS"]\n"
330 SUM=0
331 HOURS=0
332 done < "$filename"
333 }
334
335
336 #semester_courses() function used to Show or print the total number of courses taken per semester.
337 semester_courses(){
338 echo "\n"
339 #creating files for the operation
340 touch marks
341 touch hours
342 > marks
343 > hours
344 > marks_hours
345 while read line; do #read the file
346 temp=$( echo "$line" | sed -n -e 's/^.*; //p' ) #cut the lines just to show the courses and gardes
347 echo "$temp"
348 for val in $temp; do
349 val=$( echo "$val" | cut -f1 -d",") #make sure that all grades are valid
350 l=$(echo "$val" | grep [^0-9] | wc -c)
351 if [ "$l" -ge 0 -a "$l" -le 3 ]
352 then
353 if [ "$val" = "FA" ] #FA (Fail Absent counted as 50).
354 then
355 val=50
356 elif [ "$val" = "F" ] #F (Fail and counted as 55)
357 then
358 val=55
359 elif [ "$val" = "I" ] #I (incomplete)
360 then
361 val=0
362 fi
363 echo "$val" >> marks

```



```

364 else
365 echo "$val" | cut -c 6 >> hours
366 fi
367 done
368 touch marks_hours
369 paste marks hours > marks_hours
370 while read line2; do
371 temp1=$( echo "$line2" | cut -f1 )
372 temp2=$( echo "$line2" | cut -f2 )
373 if [ "$temp1" -eq 0 ]
374 then
375 continue
376 else
377 COUNT=$((COUNT+1)) # compute the total number of taken courses per semester
378 fi
379 done < marks_hours
380 > marks
381 > hours
382 > marks_hours
383 echo "\n[Total Courses in the semester: "$COUNT"]\n"
384 COUNT=0
385 done < "$filename"
386 }
387
388 #total_courses() function used to Show or print the total number of courses taken
389 total_courses(){
390 echo "\n"
391 #variables
392 COUNT=0
393 #creating files for the operation
394 touch marks
395 touch hours
396 > marks
397 > hours
398 > marks_hours
399 while read line; do #read the file
400 temp=$( echo "$line" | sed -n -e 's/^.*; //p' ) #cut the lines just to show the courses and gardes
401 for val in $temp; do
402 val=$( echo "$val" | cut -f1 -d",") #make sure that all grades and courses are valid
403 l=$(echo "$val" | grep [^0-9] | wc -c)
404 if [ "$l" -ge 0 -a "$l" -le 3 ]
405 then
406 if [ "$val" = "FA" ] #FA (Fail Absent counted as 50).
407 then
408 val=50

```

```

409 elif [ "$sval" = "F" ] #F (Fail and counted as 55)
410 then
411 val=55
412 elif [ "$sval" = "I" ] #I (incomplete)
413 then
414 val=0
415 fi
416 echo "$sval" >> marks
417 else
418 echo "$sval" | cut -c 6 >> hours
419 fi
420 done
421 done < "$filename"
422 touch marks_hours
423 paste marks hours > marks_hours
424 while read line2; do
425 temp1=$( echo "$line2" | cut -f1 )
426 temp2=$( echo "$line2" | cut -f2 )
427 if [ "$temp1" -eq 0 ]
428 then
429 continue
430 else
431 COUNT=$((COUNT+1))
432 fi
433 done < marks_hours
434 echo "\n[Total Courses: "$COUNT"]\n"
435 }
436
437 #num_of_labs() function used to Show or print the total number of labs taken.
438 num_of_labs(){
439 echo "\n"
440 #variable
441 COUNT=0
442 #creating files
443 touch marks
444 touch hours
445 > marks
446 > hours
447 > marks_hours
448 while read line; do
449 temp=$( echo "$line" | sed -n -e 's/^.*; ///

' ) #cut the lines just to show the courses and gardes
450 for val in $temp; do
451 val=$( echo "$val" | cut -f1 -d", " )
452 l=$(echo "$val" | grep [^0-9] | wc -c)
453 if [ "$l" -ge 0 -a "$l" -le 3 ]


```

```

454 then
455 if [ "$val" = "FA" ]
456 then
457 val=50
458 elif [ "$val" = "F" ]
459 then
460 val=55
461 elif [ "$val" = "I" ]
462 then
463 val=0
464 fi
465 echo "$val" >> marks
466 else
467 echo "$val" | cut -c 6 >> hours
468 fi
469 done
470 done < "$filename"
471 touch marks_hours
472 paste marks_hours > marks_hours
473 while read line2; do
474 temp1=$( echo "$line2" | cut -f1 )
475 temp2=$( echo "$line2" | cut -f2 )
476 if [ "$temp2" -ne 1 ]
477 then
478 continue
479 else
480 COUNT=$((COUNT+1)) #compute the number of labes
481 fi
482 done < marks_hours
483 echo "\n[Total Number of Labs: "$COUNT"]\n"
484 }
485
486
487
488 #record_insertion() function used for Insert the new semester record
489 record_insertion(){
490
491 while true; do
492 echo "Enter new semester in form of:-"
493 echo "Semester number [ xxxx-yyyy/(1 or 2 or 3); ]"
494 read sem
495
496
497 temp1=$( echo "$sem" | cut -c 5 ) # make sure the the giving semester is valid has (-) between the years
498 temp2=$( echo "$sem" | cut -c 10 ) #make sure the the giving semester is valid has (/) after the years
499 temp3=$( echo "$sem" | cut -c 12 ) #make sure the the giving semester is valid has (;) after the semester
500 if [ "$temp1" = "-" -a "$temp2" = "/" -a "$temp3" = ";" ] #make sure the the giving semester is valid
501 then
502 echo "Valid input"
503 break 1
504 else
505 echo "Invalid Input"
506 continue
507 fi
508 done
509
510 #elsetrf :legend
511
512 HOURS=0
513 while true; do
514
515 echo "Enter the new course [course marks] ex: ENCS3130 85 "
516 read cour_mar
517 temp1=$( echo "$cour_mar" | cut -c1-4 ) # ENEE or ENCS the first 4 char
518 temp2=$( echo "$cour_mar" | cut -c5-8 ) # the 4 numbers next to the course name
519 temp3=$( echo "$cour_mar" | cut -c6 ) # the number of hours
520 temp4=$( echo "$cour_mar" | cut -c10-11 ) # the gd
521
522
523 if [ \(( "$temp1" = "ENEE" -o "$temp1" = "ENCS" \) -a \(( "$temp2" -ge 2000 -a "$temp2" -le 5999 \) ) ] #make sure the the giving grade is valid
524 then
525 if [ "$temp4" -eq 50 ]
526 then
527 temp4=FA
528 elif [ "$temp4" -eq 55 ]
529 then
530 temp4=F
531 elif [ "$temp4" = 0 ]
532 then
533 temp4=I
534 fi
535
536 echo "Valid Course"
537 HOURS=$((HOURS+temp3)) #total hours inserted
538 echo "You have Assigned this many hours:"
539 echo $HOURS
540
541 echo "Do you want to enter another course? (y/n)"
542 read input
543 else

```

```

544 echo "Invalid Course"
545 continue
546 fi
547
548 if [ "$input" = y -a "$HOURS" -le 18 ] #insert anothe course if y pressed
549 then
550 sem="$sem" "${temp1}" "${temp2}" "${temp4}," #concat the valid gd to the semester
551 continue
552 elif [ "$input" = n -a "$HOURS" -lt 12 ] #end of the inserion but if the total hours less than 12 print he hours you have are less than 12 therefore you have to add more hours
553 then
554 echo "The hours you have are less than 12 therefore you have to add more hours:"
555 sem="$sem" "${temp1}" "${temp2}" "${temp4},"
556 continue
557 elif [ "$input" = n -a "$HOURS" -ge 12 ] #concat the last inserted course and append the semester to the file
558 then
559 sem="$sem" "${temp1}" "${temp2}" "${temp4}"
560 echo "$sem" >> "$filename"
561 break
562 elif [ "$HOURS" -gt 18 ] #if the number of hours greater than 18
563 then
564 sem="$sem" "${temp1}" "${temp2}" "${temp4}"
565 echo "$sem" >> "$filename"
566 break
567 elif [ "$input" != y -o "$input" = n ] #make sure that the answer should be y for yes or n fo no
568 then
569 echo "Invalid answer"
570 break
571 fi
572 done
573 }
574
575
576 #change_grade() function used to change in course grade by giving the course name
577 change_grade(){
578 echo "Enter the course id that you want to change"
579 flag=0
580 read chg
581 touch marks
582 touch hours
583 > marks
584 > hours
585 > marks_hours
586 while read line; do
587 temp=$( echo "$line" | sed -n -e 's/^.*: //p' ) #cut the lines just to show the courses and gardes
588 echo "$temp"
589 for val in $temp; do #read the courses and the grades and saves echa in file
590 val=$( echo "$val" | cut -f1 -d",," )
591 l=$(echo "$val" | grep [^0-9] | wc -c)
592
593 if [ "$l" -ge 0 -a "$l" -le 3 ]
594 then
595 echo "$val" >> marks
596 else
597 echo "$val" >> hours
598 fi
599
600 done
601 done < "$filename"
602 touch marks_hours
603 paste hours marks > marks_hours
604 while read line2; do #see if the entered course is exist or not
605 temp1=$( echo "$line2" | cut -f1 )
606 temp2=$( echo "$line2" | cut -f2 )
607
608 if [ "$temp1" = "$chg" ]
609 then
610 echo "$line2"
611 flag=1
612 break
613 else
614 continue
615 fi
616 done < marks_hours
617
618 if [ "$flag" -eq 1 ]
619 then
620 echo "Enter new grade"
621 read gd
622 echo "$gd"
623
624 n=$(echo "$grade" | grep [^0-9] | wc -c)
625
626 if [ "$n" -ge 0 -a "$n" -le 3 ]
627 then
628 echo "Do you to change the grade? (y/n)"
629 read ans
630 if [ "$ans" = "y" ]
631 then
632 sed -e "s/$chg [0123456789F][0123456789A]/$chg $gd/g" "$filename" > tempfile # the changing operation fot the grade
633 cat tempfile > "$filename"

```

```

634 elif [ "$ans" = "n" ]
635 then
636 return
637
638 else
639 echo "Invalid answer"
640 fi
641
642 else
643 continue
644 fi
645 else
646 echo "Invalid input"
647 fi
648 }
649
650
651
652
653
654
655 echo Enter File Name
656 read filename
657 if [ ! -e "$filename" ] #see if the file exist or not
658 then
659     echo "There is no such File"
660     exit 0
661 fi
662
663 while true
664 do
665 #the list we has the operations description on it
666
667 echo "1. print student records (all semesters)."
668 echo "2. print student records for a specific semester."
669 echo "3. print the overall average."
670 echo "4. print the average for every semester."
671 echo "5. print the total number of passed hours."
672 echo "6. print the percentage of total passed hours in relation to total F and FA hours."
673 echo "7. print the total number of hours taken for every semester."
674 echo "8. print the total number of courses taken."
675 echo "9. print the total number of labs taken."
676 echo "10. Insert the new semester record."
677 echo "11. Change in course grade."
678 echo "0. Exit Program"
679
680 read -p 'Enter your command: ' command
681 case "$command" in
682 1)
683 #case 1 to print student records (all semesters).
684 echo "\n"
685 cat "$filename"
686 echo "\n";;
687 2)
688 #case 2 to print student records for a specific semester. by calling print_specific_semester function
689 print_specific_semester;;
690 3)
691 #case 3 to print the overall average. by calling print_overall_avg function
692 print_overall_avg;;
693 4)
694 #case 4 to print the average for every semester. by calling print_semester_avg function
695 print_semester_avg;;
696 5)
697 #case 5 to print the total number of passed hours. by calling total_passed_hours function
698 total_passed_hours;;
699 6)
700 #case 6 to Show or print the percentage of total passed hours in relation to total F and FA hours. by calling percentage_of_passed_hours function
701 percentage_of_passed_hours;;
702 7)
703 #case 7 to print the total number of hours taken for every semester. by calling hours_taken function
704 hours_taken;;
705 8)
706 #case 8 to print the total number of courses taken. by calling semester_courses total_courses functions
707 semester_courses
708 total_courses
709 ;;
710 9)
711 #case 9 to print the total number of labs taken. by calling num_of_labs functions
712 num_of_labs;;
713 10)
714 #case 10 to Insert the new semester record. by calling record_insertion function
715 record_insertion;;
716 11)
717 #case 11 Change in course grade. by calling change_grade function
718 change_grade;;
719 0)
720 echo "Thank you for using" #case 0 (exit case)
721 exit 0;;
722 *)
723 echo "\ninvalid input\n\n" #default case for any invalid commands
724 esac
725 done

```

Examples:-

```
nidal@nidal:~$ ./studentRecord
Enter File Name
jksdfh
There is no such File
nidal@nidal:~$ ./studentRecord
Enter File Name
Record
1. print student records (all semesters).
2. print student records for a specific semester.
3. print the overall average.
4. print the average for every semester.
5. print the total number of passed hours.
6. print the percentage of total passed hours in relation to total F and FA hours.
7. print the total number of hours taken for every semester.
8. print the total number of courses taken.
9. print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
0. Exit Program
Enter your command: 1

2021-2022/1; ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88
2021-2022/2; ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88
2022-2023/1; ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78
2022-2023/2; ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78
```

```
1. print student records (all semesters).
2. print student records for a specific semester.
3. print the overall average.
4. print the average for every semester.
5. print the total number of passed hours.
6. print the percentage of total passed hours in relation to total F and FA hours.
7. print the total number of hours taken for every semester.
8. print the total number of courses taken.
9. print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
0. Exit Program
Enter your command: 2
Enter the semester
2022-2023/3

There is no such semester
```

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

Enter your command: 2

Enter the semester

2022-2023/1

2022-2023/1; ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78

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

Enter your command: 3

ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88
ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88
ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78
ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78

The Total Points: 3540

The Total Hours: 47

```
-----
|                                     |
|           The overall average: 75   |
|                                     |
|-----|
```

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

Enter your command: 4

ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88

The Total Points: 4734

The Total Hours: 63

```
-----
|                                     |
|           The overall average: 75   |
|                                     |
|-----|
```

ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88

The Total Points: 746

The Total Hours: 9

```
-----
|                                     |
|           The overall average: 82   |
|                                     |
|-----|
```

ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78

The Total Points: 741

The Total Hours: 10

```
-----
|                                     |
|           The overall average: 74   |
|                                     |
|-----|
```


The overall average: 71

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

Total Passed Hours: 41

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

Failed Hours: 6

Passed Hours: 41

```
-----
|                                     |
|      Percentage of Failed Hours: 14%      |
|                                     |
|-----|
|                                     |
|      Percentage of PASSED Hours: 86%      |
|                                     |
|-----|
```

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

Enter your command: 7

ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88

[Number of Hours Taken: 16]

ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88

[Number of Hours Taken: 9]

ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78

[Number of Hours Taken: 10]

ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78

[Number of Hours Taken: 12]

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

```
ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88
```

```
[Total Courses in the semester: 5]
```

```
ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88
```

```
[Total Courses in the semester: 4]
```

```
ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78
```

```
[Total Courses in the semester: 5]
```

```
ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78
```

```
[Total Courses in the semester: 5]
```

```
[Total Courses: 19]
```

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

```
[Total Number of Labs: 2]
```

```

1. print student records (all semesters).
2. print student records for a specific semester.
3. print the overall average.
4. print the average for every semester.
5. print the total number of passed hours.
6. print the percentage of total passed hours in relation to total F and FA hours.
7. print the total number of hours taken for every semester.
8. print the total number of courses taken.
9. print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
0. Exit Program
Enter your command: 10
Enter new semester in form of:-
Semester number [ xxxx-yyyy/(1 or 2 or 3); ]
5456-46484/4648
Invalid Input
Enter new semester in form of:-
Semester number [ xxxx-yyyy/(1 or 2 or 3); ]
2023-2024/1;
Valid input
Enter the new course [course marks] ex: ENCS3130 85
ENEN2455
Invalid Course
Enter the new course [course marks] ex: ENCS3130 85
ENEE3698 55
Valid Course
You have Assigned this many hours:
6
Do you want to enter another course? (y/n)
y
Enter the new course [course marks] ex: ENCS3130 85
ENEE4156 77
Valid Course
You have Assigned this many hours:
7
Do you want to enter another course? (y/n)
n
The hours you have are less than 12 therefore you have to add more hours:
Enter the new course [course marks] ex: ENCS3130 85
ENCS5234 88
Valid Course
You have Assigned this many hours:
9
Do you want to enter another course? (y/n)

```

```

Do you want to enter another course? (y/n)
y
Enter the new course [course marks] ex: ENCS3130 85
ENCS3574 85
Valid Course
You have Assigned this many hours:
14
Do you want to enter another course? (y/n)
n
1. print student records (all semesters).
2. print student records for a specific semester.
3. print the overall average.
4. print the average for every semester.
5. print the total number of passed hours.
6. print the percentage of total passed hours in relation to total F and FA hours.
7. print the total number of hours taken for every semester.
8. print the total number of courses taken.
9. print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
0. Exit Program
Enter your command: 1

2021-2022/1; ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88
2021-2022/2; ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88
2022-2023/1; ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78
2022-2023/2; ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78
2023-2024/1; ENEE3698 F, ENEE4156 77, ENCS5234 88, ENCS3574 85

```

```

1. print student records (all semesters).
2. print student records for a specific semester.
3. print the overall average.
4. print the average for every semester.
5. print the total number of passed hours.
6. print the percentage of total passed hours in relation to total F and FA hours.
7. print the total number of hours taken for every semester.
8. print the total number of courses taken.
9. print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
0. Exit Program
Enter your command: 11
Enter the course id that you want to change
ENCS2323
ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88
ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88
ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78
ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78
ENEE3698 F, ENEE4156 77, ENCS5234 88, ENCS3574 85
Invalid input

```

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

Enter your command: 11

Enter the course id that you want to change

ENCS3232

```
ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88
ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88
ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 80, ENCS2235 78
ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78
ENEE3698 F, ENEE4156 77, ENCS5234 88, ENCS3574 85
```

ENCS3232 80

Enter new grade

90

90

Do you to change the grade? (y/n)

y

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

Enter your command: 1

```
2021-2022/1; ENEE2304 85, ENCS2300 FA, ENCS3320 F, ENCS2442 90, ENEE2365 88
2021-2022/2; ENEE2224 90, ENCS2360 I, ENCS3220 60, ENCS2332 90, ENEE2235 88
2022-2023/1; ENCS2224 70, ENEE2360 75, ENCS2120 60, ENCS3232 90, ENCS2235 78
2022-2023/2; ENCS2334 70, ENEE2160 75, ENCS2320 60, ENCS2232 80, ENCS2335 78
2023-2024/1; ENEE3698 F, ENEE4156 77, ENCS5234 88, ENCS3574 85
```

```
1. print student records (all semesters).
2. print student records for a specific semester.
3. print the overall average.
4. print the average for every semester.
5. print the total number of passed hours.
6. print the percentage of total passed hours in relation to total F and FA hours.
7. print the total number of hours taken for every semester.
8. print the total number of courses taken.
9. print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
0. Exit Program
Enter your command: 0
Thank you for using
```

```
1. print student records (all semesters).
2. print student records for a specific semester.
3. print the overall average.
4. print the average for every semester.
5. print the total number of passed hours.
6. print the percentage of total passed hours in relation to total F and FA hours.
7. print the total number of hours taken for every semester.
8. print the total number of courses taken.
9. print the total number of labs taken.
10. Insert the new semester record.
11. Change in course grade.
0. Exit Program
Enter your command: 55
```

invalid input

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

invalid input