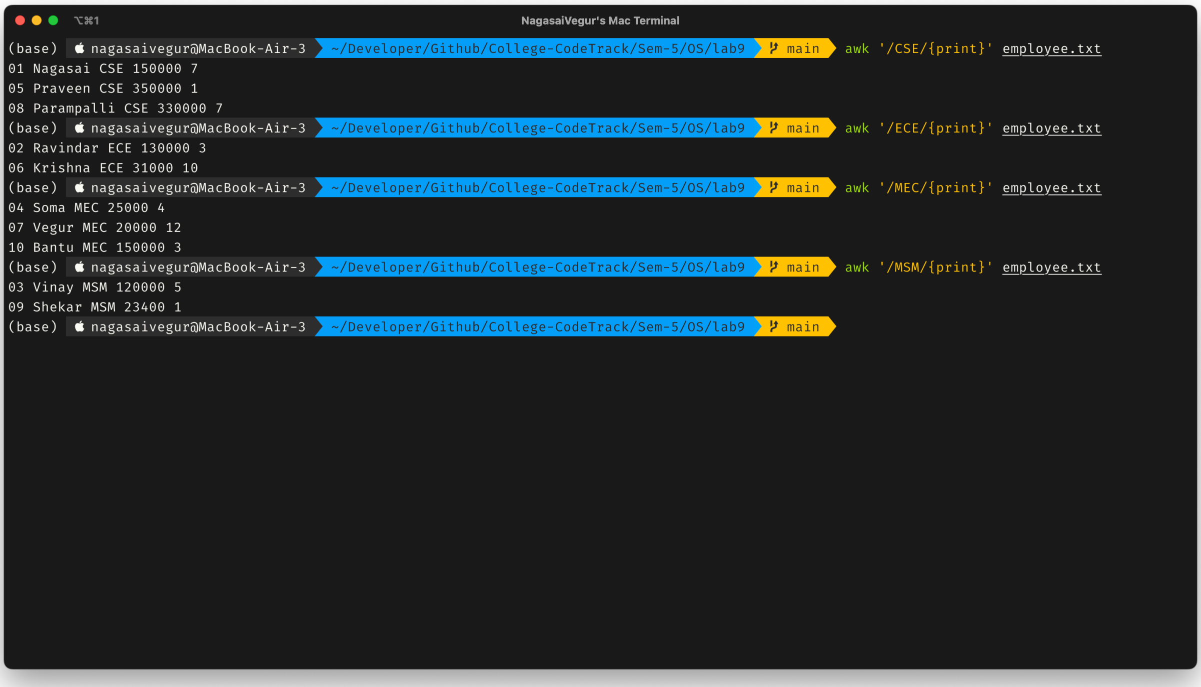
Operating System Lab 9

awk and grep commands implementation.

1. Create a text file with employee details like EmpID, EmpName, Department, Salary, Years of experience (Use awk).

a. Print the rows of employees who belong to a specific Department.

- We need to print details of employee belonging to a specific department.
  In the below command replace Department with the specific department.
  awk '/Department/{print}' employee.txt

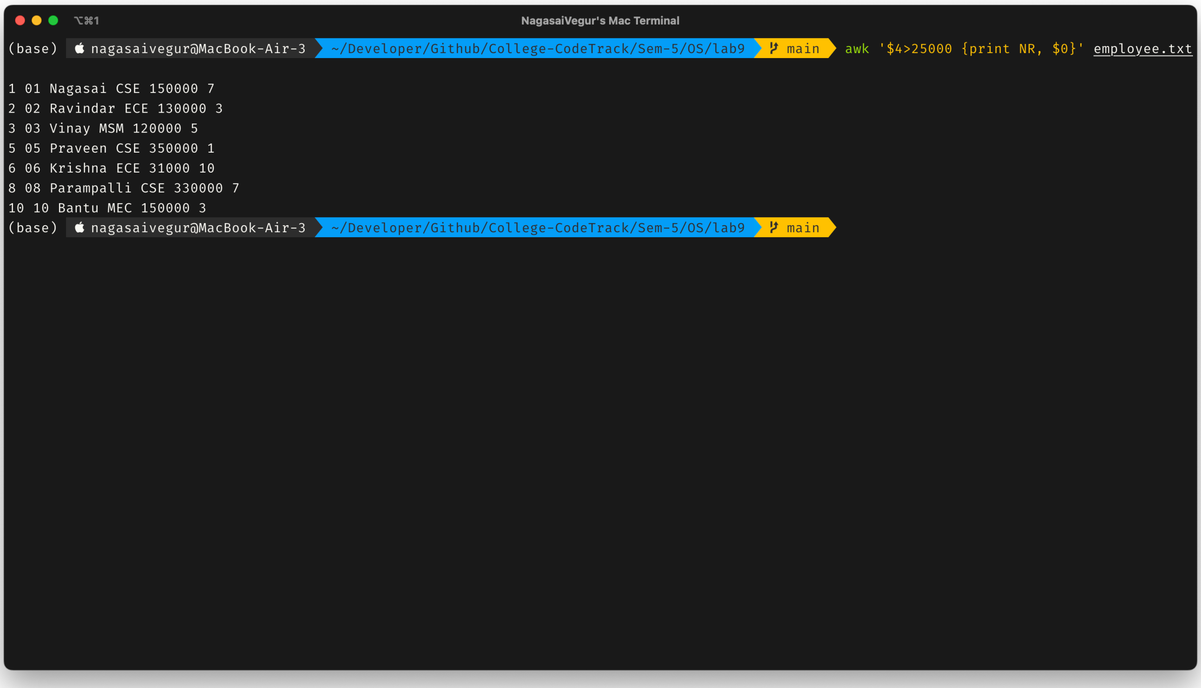


```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk '/CSE/{print}' employee.txt
01 Nagasai CSE 150000 7
05 Praveen CSE 350000 1
08 Parampalli CSE 330000 7
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk '/ECE/{print}' employee.txt
02 Ravindar ECE 130000 3
06 Krishna ECE 31000 10
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk '/MEC/{print}' employee.txt
04 Soma MEC 25000 4
07 Vegur MEC 20000 12
10 Bantu MEC 150000 3
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk '/MSM/{print}' employee.txt
03 Vinay MSM 120000 5
09 Shekar MSM 23400 1
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main
```

b. Print the employee details whose salary is above 25000 with line numbers.

- awk '$4>25000 {print NR, $0}' employee.txt

Here $4 gives the 4th column, i.e., salary column and searches on it for greater than 25000 value. And in the printing NR represents the line number and $0 represents all the details of that current row.



NagasaiVegur's Mac Terminal

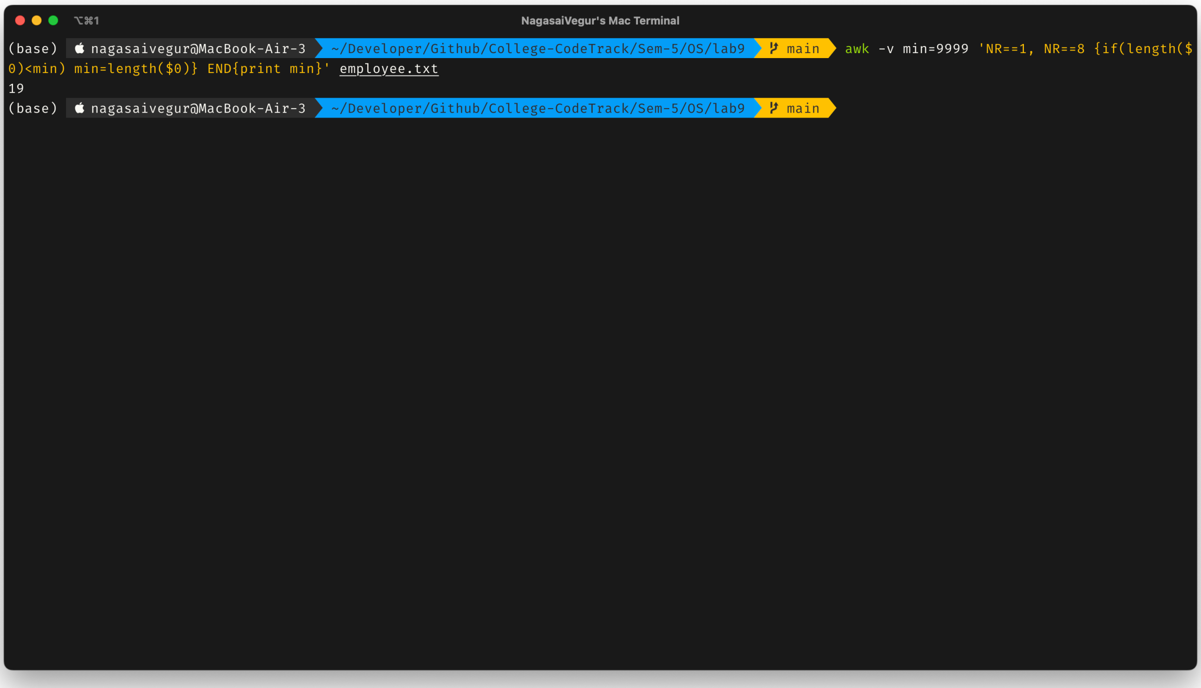
```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk '$4>25000 {print NR, $0}' employee.txt

1 01 Nagasai CSE 150000 7
2 02 Ravindar ECE 130000 3
3 03 Vinay MSM 120000 5
5 05 Praveen CSE 350000 1
6 06 Krishna ECE 31000 10
8 08 Parampalli CSE 330000 7
10 10 Bantu MEC 150000 3
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main
```

c. Print length of the minimum row in the specific range (use NR for Row Number).

- awk -v min=9999 ‘NR==1 NR==8 {if(length($0) < min) min=length($0)} END {print min}’ employee.txt

Here -v used to avoid misprints. NR==1 NR==8 chooses a specific range between row 1 to row 8. min=9999 sets the initial value of min. Then based on the if condition on all the rows it is looped over and updates the min if concurred a minimum than the previous. and after END of the loop prints the minimum value.



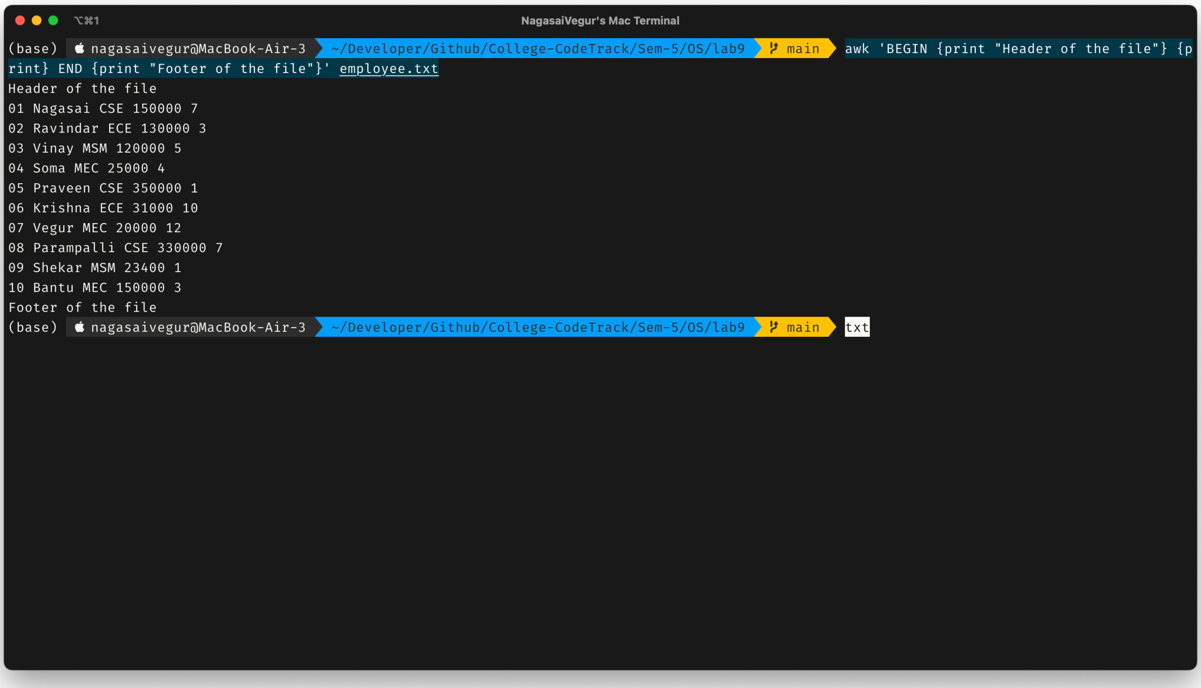
NagasaiVegur's Mac Terminal

```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk -v min=9999 'NR==1, NR==8 {if(length($
0)<min) min=length($0)} END{print min}' employee.txt
19
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main
```

d. Print employee details with Header and Footer.

- awk 'BEGIN {print "Header of the file"} {print} END {print "Footer of the file"}' employee.txt

Here BEGIN and END manages the start and end of table while print manages the everything in that given file.

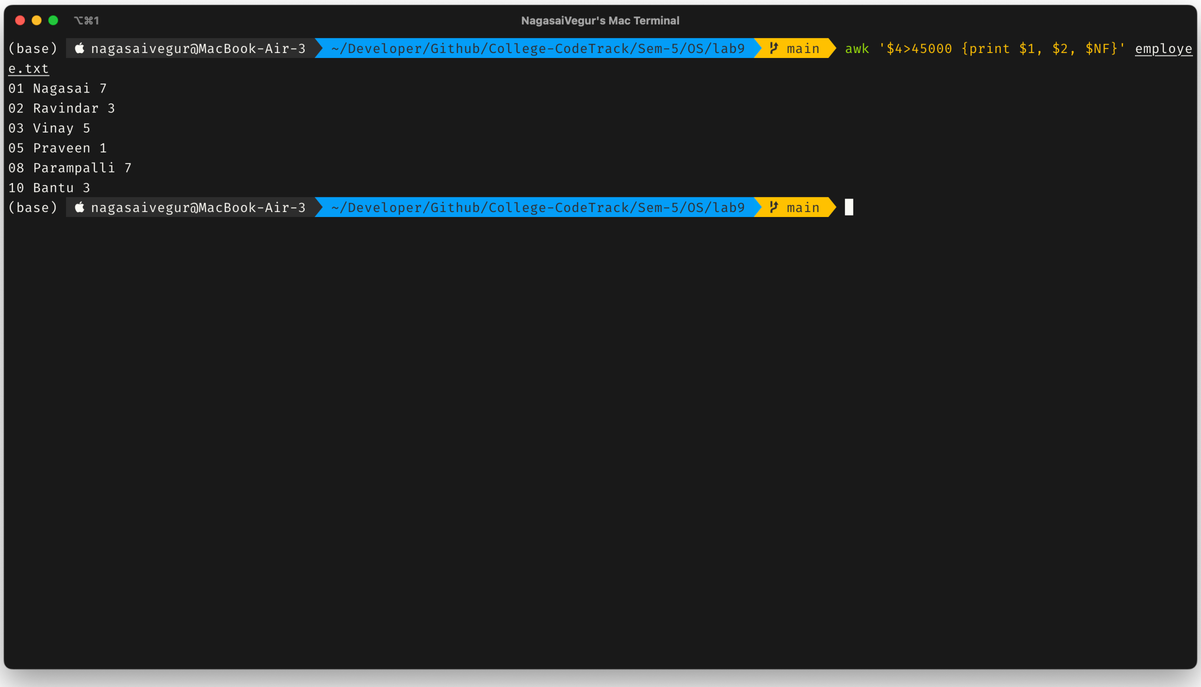


```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk 'BEGIN {print "Header of the file"} {p
rint} END {print "Footer of the file"}' employee.txt
Header of the file
01 Nagasai CSE 150000 7
02 Ravindar ECE 130000 3
03 Vinay MSM 120000 5
04 Soma MEC 25000 4
05 Praveen CSE 350000 1
06 Krishna ECE 31000 10
07 Vegur MEC 20000 12
08 Parampalli CSE 330000 7
09 Shekar MSM 23400 1
10 Bantu MEC 150000 3
Footer of the file
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  txt
```

e. Print Employee details, whose salary is greater than 45000, print first, second and last column of the file.

- awk '$4>45000 {print $1, $2, $NF}' employee.txt

$4 represents salary column and based on the condition greater than 45000, prints $1 - first column, $2 - second column and $NF - number of fields i.e., last column.

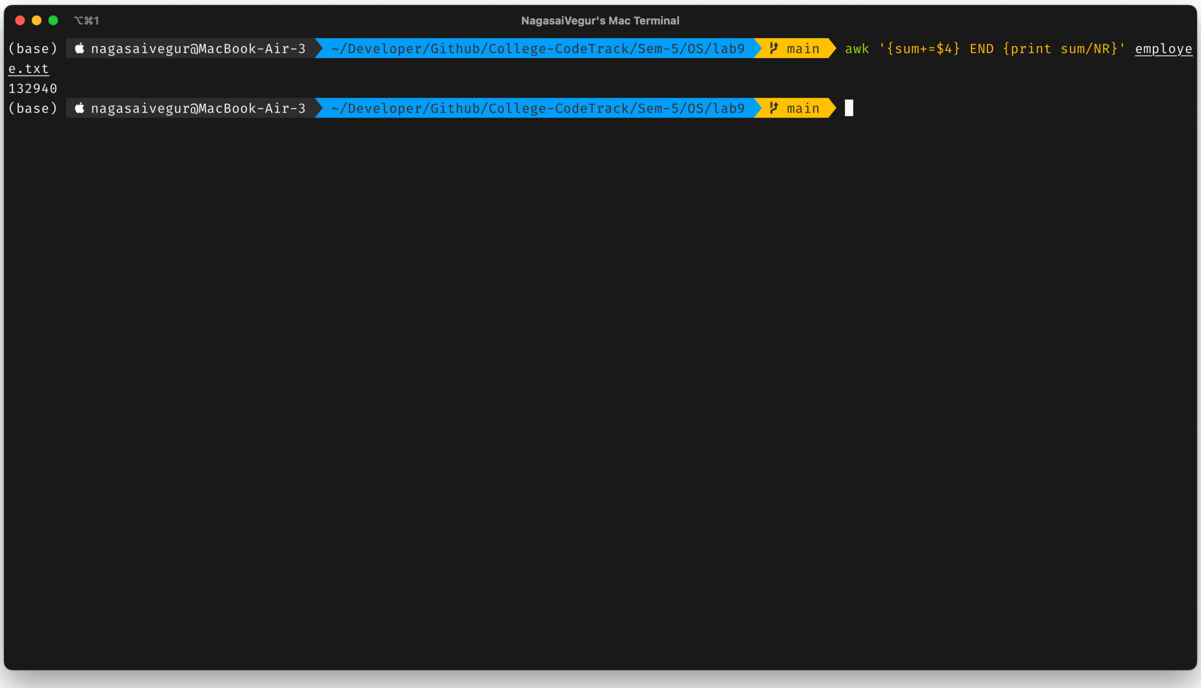


```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk '$4>45000 {print $1, $2, $NF}' employe
e.txt
01 Nagasai 7
02 Ravindar 3
03 Vinay 5
05 Praveen 1
08 Parampalli 7
10 Bantu 3
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main 
```

f. Print average salary of the employee.

- awk '{sum+=$4} END {print sum/NR}' employee.txt

Here {sum+=4} code block loops over every row and sums up 4th column value i.e., salary and at the END prints sum/NR. NR represents number of rows at the end.

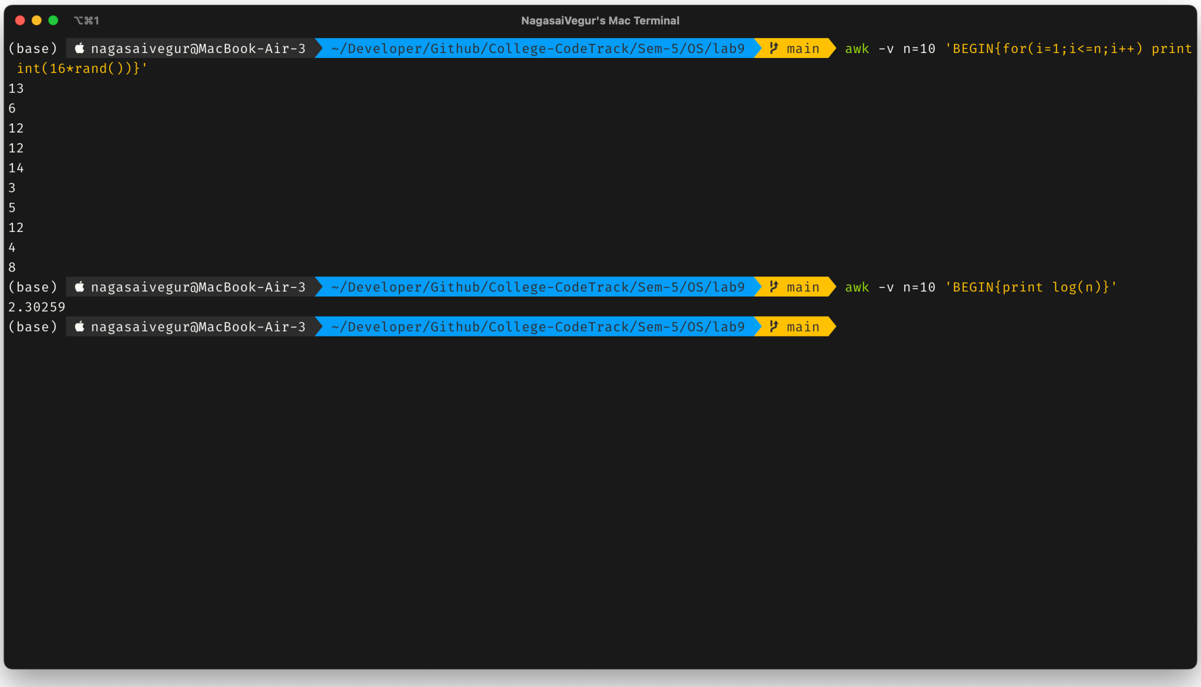


```
(base)  nagasaivegur@MacBook-Air-3 ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk '{sum+=$4} END {print sum/NR}' employe
e.txt
132940
(base)  nagasaivegur@MacBook-Air-3 ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main 
```

2) Use awk command to

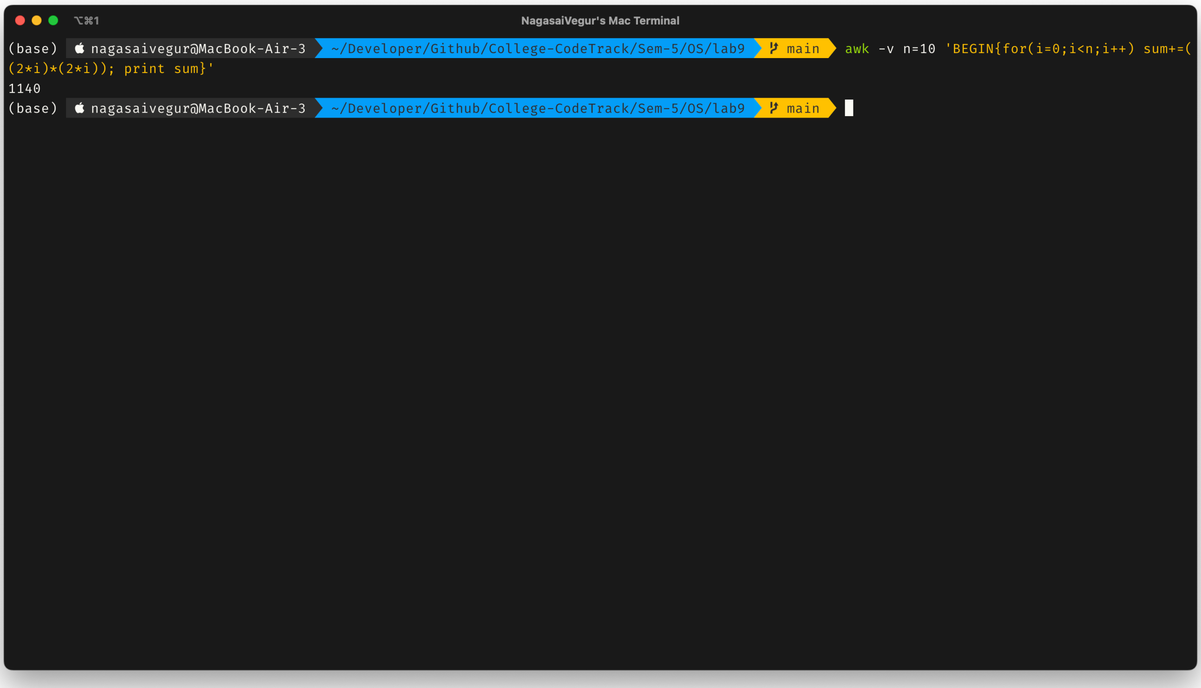
a. Print “n” random numbers and print log(n) using the in-built function

- printing 10 random numbers with some seed 16(random) and log of n using built in function.



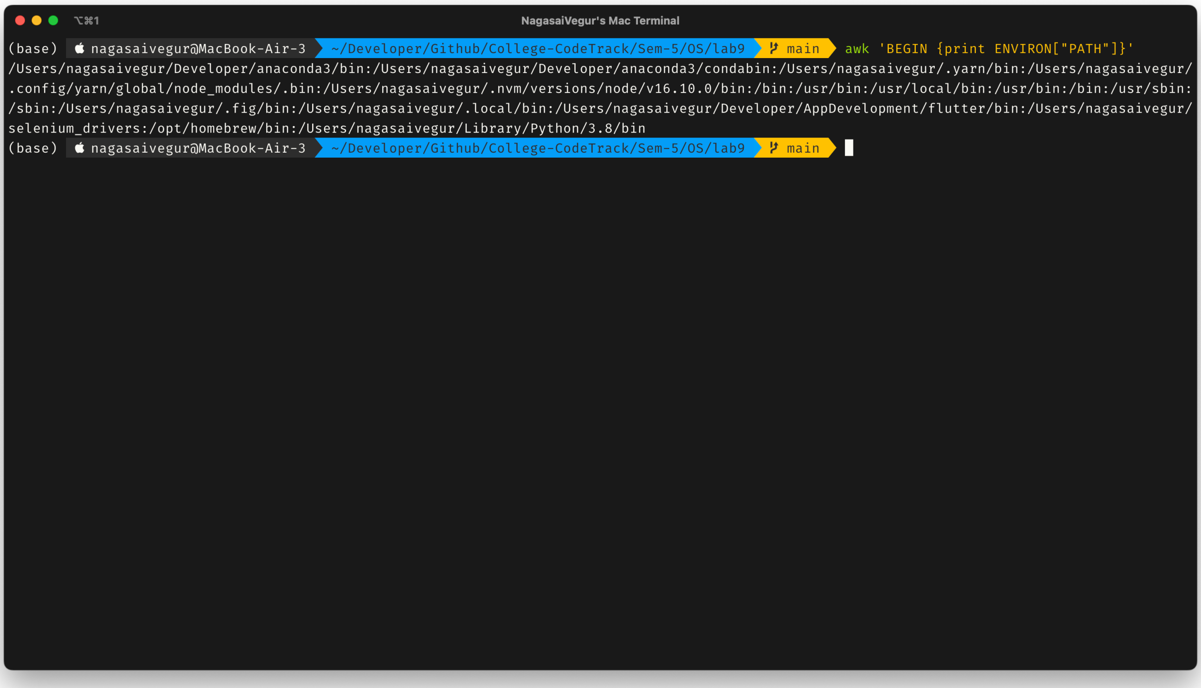
```
(base)  nagasaivegur@MacBook-Air-3 ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk -v n=10 'BEGIN{for(i=1;i<=n;i++) print
 int(16*rand())}'
13
6
12
12
14
3
5
12
4
8
(base)  nagasaivegur@MacBook-Air-3 ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk -v n=10 'BEGIN{print log(n)}'
2.30259
(base)  nagasaivegur@MacBook-Air-3 ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main 
```

b. Print sum of squares of first “n” even numbers.



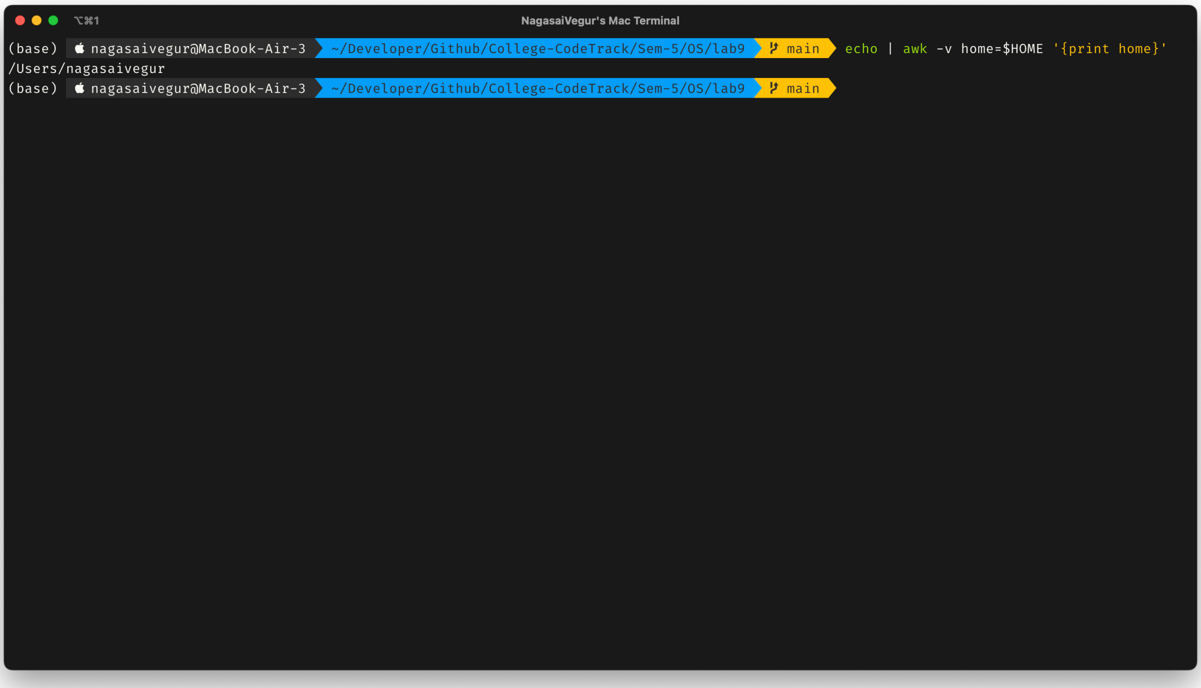
```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk -v n=10 'BEGIN{for(i=0;i<n;i++) sum+=(
(2*i)*(2*i)); print sum}'
1140
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main 
```

c. Print Environmental variables set on Machine.



```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  awk 'BEGIN {print ENVIRON["PATH"]}'
/Users/nagasaivegur/Developer/anaconda3/bin:/Users/nagasaivegur/Developer/anaconda3/condabin:/Users/nagasaivegur/.yarn/bin:/Users/nagasaivegur/
.config/yarn/global/node_modules/.bin:/Users/nagasaivegur/.nvm/versions/node/v16.10.0/bin:/bin:/usr/bin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:
/sbin:/Users/nagasaivegur/.fig/bin:/Users/nagasaivegur/.local/bin:/Users/nagasaivegur/Developer/AppDevelopment/flutter/bin:/Users/nagasaivegur/
selenium_drivers:/opt/homebrew/bin:/Users/nagasaivegur/Library/Python/3.8/bin
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main 
```

d. Print path of “Home”



```
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main  echo | awk -v home=$HOME '{print home}'
/Users/nagasaivegur
(base)  nagasaivegur@MacBook-Air-3  ~/Developer/Github/College-CodeTrack/Sem-5/OS/lab9  main
```

1. **Create three files with text and Use grep t**

a. Print lines from all the files with matching lines with ignored cases.**h is used to print matched lines i is used to ignore the case**

grep -hi “hello” text1.txt text2.txt text3.txt

e. grep -C 3 “hello” text1.txt

f. grep -vi “hello” text1.txt