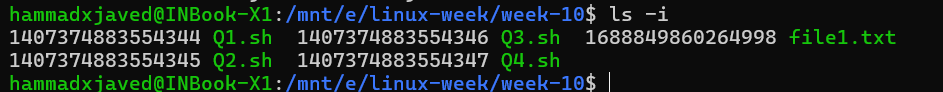
Week 9

1. Write a report on the structure of the Linux file system.

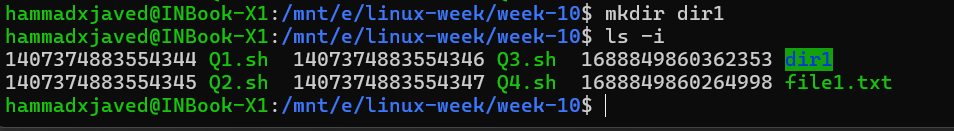
The Linux file system is a hierarchical directory structure that starts from the root directory (/). Here are the key components:

* **Root Directory (**/**)**: The top-level directory.
* **/bin**: Essential command binaries.
* **/boot**: Boot loader files.
* **/dev**: Device files.
* **/etc**: Configuration files.
* **/home**: User home directories.
* **/lib**: Essential shared libraries.
* **/media**: Mount points for removable media.
* **/mnt**: Temporary mount points.
* **/opt**: Optional software packages.
* **/proc**: Process and kernel information.
* **/root**: Home directory for the root user.
* **/sbin**: System binaries.
* **/tmp**: Temporary files.
* **/usr**: User binaries and read-only data.
* **/var**: Variable data files.

2. Display inode information using ls -i and interpret the results.



3. Create and delete files and directories, and observe changes in inode numbers.



4. Explain the significance of inodes in file management and demonstrate with examples.

Inodes store metadata about files, such as:

* File size
* Permissions
* Owner and group
* Timestamps
* Data block pointers

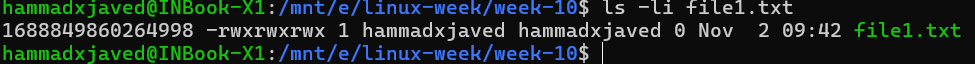
Example:

ls -li file1.txt

Output:

1234567 -rw-r--r-- 1 user group 0 Nov 15 22:19 file1.txt

The inode number 1234567 stores all metadata except the filename.



5. Write a program in python to find alphabet/s having maximum number of instances in a given file.

def max\_alphabet\_instances(file\_path):

    with open(file\_path, 'r') as file:

        content = file.read().lower()

    alphabet\_count = {}

    for char in content:

        if char.isalpha():

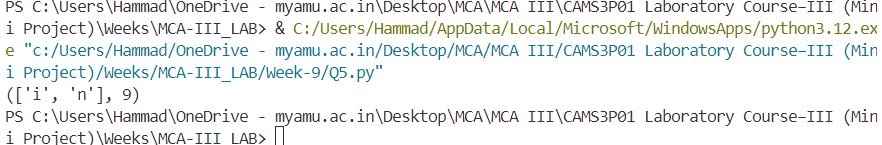
            alphabet\_count[char] = alphabet\_count.get(char, 0) + 1

    max\_count = max(alphabet\_count.values())

    max\_alphabets = [char for char, count in alphabet\_count.items() if count == max\_count]

    return max\_alphabets, max\_count

print(max\_alphabet\_instances('Week-9/file.txt'))



6. A file contains information about programs and courses in the following format:

Program,course. Write a Python program to find the number of courses against each program.

Eg:

Program,Course

MCA,Database

MCA,Java

M.Sc,Data Structure

B.Sc, Python

def count\_courses(file\_path):

course\_count = {}

with open(file\_path, 'r') as file:

for line in file:

program, course = line.strip().split(',')

if program in course\_count:

course\_count[program] += 1

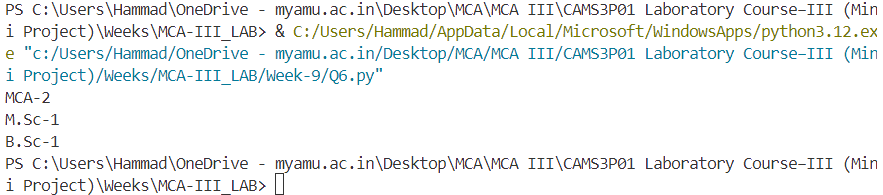
else:

course\_count[program] = 1

for program, count in course\_count.items():

print(f"{program}-{count}")

count\_courses('Week-9/programs.csv')



7. A file contains information about employees with the following parameters: Name, Id, Salary, Dname. Write a Python program to write one more column HRA (House rent allowances) to this file, where HRA= 18%of salary

Eg: Suppose the existing file is as follows, where you need to add HRA column:

Name,id,salary, Dname

Amar,101,20000,Sales

Ammar,102,22000,Marketing

Rahil,103,18000,Sales

def add\_hra\_column(file\_path):

    with open(file\_path, 'r') as file:

        lines = file.readlines()

    with open('updated\_employees.csv', 'w') as file:

        for line in lines:

            name, emp\_id, salary, dname = line.strip().split(',')

            hra = round(0.18 \* float(salary), 2)

            file.write(f"{name},{emp\_id},{salary},{dname},{hra}\n")

add\_hra\_column('Week-9/employees.csv')

