**NAAN MUDHALVAN**

**CHALLENGE 3**

**Challenge 1**

Write a function called linear\_search\_product that takes the list of products and a

target product name as input. The function should perform a linear search to find the

target product in the list and return a list of indices of all occurrences of the product If

found, or an empty list if the product is not found.

**PROGRAM:**

def linear\_search\_product(product\_list, target\_product):

indices = []

for i, product in enumerate(product\_list):

if product == target\_product:

indices.append(i)

return indices

# Example usage:

products = ["apple", "banana", "apple", "orange", "grape", "apple"]

target = "apple"

result = linear\_search\_product(products, target)

if result:

print(f"The product '{target}' was found at indices: {result}")

else:

print(f"The product '{target}' was not found in the list.")

**Challenge 2**

Implement a function called sort\_students that takes a list of student objects as input

and sorts the list based on their CGPA (Cumulative Grade Point Average) in

descending order. Each student object has the following attributes: name (string),

rolLnumber (string), and cgpa (float). Test the function with different input lists of

students.

**PROGRAM:**

class Student:

def \_\_init\_\_(self, name, roll\_number, cgpa):

self.name = name

self.roll\_number = roll\_number

self.cgpa = cgpa

def \_\_str\_\_(self):

return f"{self.name} (Roll Number: {self.roll\_number}, CGPA: {self.cgpa})"

def sort\_students(student\_list):

# Sort the student\_list in descending order based on CGPA

sorted\_students = sorted(student\_list, key=lambda student: student.cgpa, reverse=True)

return sorted\_students

# Example usage:

students = [

Student("Alice", "A101", 3.8),

Student("Bob", "B102", 3.9),

Student("Charlie", "C103", 3.7),

Student("David", "D104", 3.95),

Student("Eve", "E105", 3.85),

]

sorted\_students = sort\_students(students)

# Printing the sorted list of students

for student in sorted\_students:

print(student)