HORIZON SCHOOL OF DIGITAL TECHNOLOGIES	**Lab 2** **Advanced Algorithm and Complexity (Python)**	**M1** **Horizon School of Digital Technologies** **2024-2025** **Rania Yangui**

Lab 2 : Searching/Sorting Programs, Managing Files

Exercise 1 :

The goal of this exercise is to master file handling and list manipulation. Write a Python program that performs the following tasks:

1) Create a text file. This file will contain one word per line (the words are provided by the user).
2) Read the file and build a list with all these words.
3) Verify that the list loaded in the previous step is sorted.
4) Search for a word X in the list and return its position or indicate that the word is not found. Use a function that takes two parameters: the list and the word to search for. It should return an integer. To check if two strings are equal, use the == operator.
5) Use merge sort to sort the list.
6) Save the contents of the sorted list back into the file.
7) Perform a binary search for a word in the sorted list.
8) What is, at worst, the cost of a non-binary search?
9) What is, at worst, the cost of a binary search?

Exercise 2 :

You are tasked with developing an order management system for a restaurant. This system should allow you to track customer orders, manage menus, calculate totals, and generate reports. To do this, you will manipulate various data structures, such as lists, tuples, dictionaries, and sets, to efficiently manage the different tasks.

Menu Management

1) Create a data structure to store the names of dishes and their prices.
2) Add, modify, and remove dishes from the menu.
3) Display the updated menu after each operation.

Order Management

Each customer order will be represented by a data structure containing the customer's name and a list of ordered dishes. Once created, this order should not be modified.

4) Create an order for a customer (name and dishes) and add it to a set of orders while avoiding duplicates.
5) Check if a specific order exists in the set of orders.
6) For each order, calculate the total by adding the prices of the ordered dishes.

Payment Management

7) Use a data structure to track the payment status of customers. Associate each customer with a status indicating whether their payment has been made.
8) Add a function to update the payment status of a customer after receiving their payment.
9) Display the current payment status for all customers.

Sales Reports

10) Store the number of times each dish has been ordered.
11) Create a function to generate a sales report, indicating the most popular dishes.