**Ministry of Higher Education and Scientific Research**
**UNIVERSITE M'Hamed BOUGARA –BOUMERDES**
**FACULTY OF SCIENCE - DEPARTMENT OF COMPUTER SCIENCE**

**Nature of the document : Tutorial 2** **Course : ASD3** **Année : 2024-2025**

**Exercise 1**

Let be two simple linear lists of integers with heads D1 and D2.
Write a procedural C++ program with functions to concatenate these two lists.

a- Without sacrificing the original lists
b- By sacrificing the original lists

**Exercise 2**

Let be a simple linear list of N integers with head D1 (N even).
Write a procedural C++ program with functions to partition this list into two linear lists of the same size with heads D2 and D3.

a- Without sacrificing the original list
b- By sacrificing the original list

**Exercise 3**

Repeat questions 1 and 2 using bidirectional linear lists.

**Exercise4**

Write a procedural C++ function to sort a simple linear list of integers.

**Exercise 5**

To represent two polynomials, we consider two linear lists of heads P1 and P2, where each node contains an exponent and the corresponding coefficient.
Assuming that the lists are sorted in descending order of powers, write a procedural C++ function and its call to construct the list representing the sum of the two polynomials.

**Exercise 6**

Let be a linear list **L** containing the description of processes (programs) in a multi-user system. Each process is described by a structure with an **id** (integer) and a **priority** (integer):

```
Struct process {int id; int prior; process* next} ;
```

Write a procedural C++ program with functions to do the following:

1. Construct a dynamic array of M linear lists, each composed of N processes.
2. Display all lists.
3. Reverse all lists.
4. Display all lists after the reversal.
5. Remove from all lists in the array, the first process of equal priority to a given value 'val' if it exists.
6. Display the lists after the delete operation.