Higher School of Computer and Digital Sciences and Technologies **2024/2025**
ADDS (Algorithmic 2) **1st year Preparatory Class**

Lab Serie No. 03: Queues & Stacks

Declaration Syntax

Linked list-based implementation

```
/* Definition of a node structure */
typedef struct node{
    <Type> data;
    Struct node* next;
} Node;
```

```
/* The queue structure */
typedef struct {
    Node *front;
    Node *rear;
} Queue;
```

```
/* The stack structure */
typedef struct {

    Node *top;

} Stack;
```

Array-based implementation

```
/* The queue structure */
#define MAX_SIZE 100
typedef struct {
    <Type> items[MAX_SIZE]

      int   front, rear;

} Queue ;
```

```
/* The Stack structure */
#define MAX_SIZE 100
typedef struct {
    <Type> items[MAX_SIZE]

    int   top;

} Stack;
```

Exercise 01

1. Write subprograms of basic operations to handle a queue implemented by an array.
2. Write subprograms of basic operations to handle a queue implemented by a linked list.

Exercise 02

A priority queue is a collection of items where each new element is inserted according to its priority, and removed from the head.

- Implement the corresponding model (basic operations) using a linear linked list.
- Write a subprogram that counts the number of elements in the queue.
- Write a subprogram that displays the elements of the queue.

Exercise 03

1. Write subprograms of basic operations to handle a stack implemented by an array.
2. Write subprograms of basic operations to handle a stack implemented by a linked list.

Exercise 04

Consider a stack S1 of positive integers.

- Write a Subprogram to move the integers from S1 into a stack S2 so that in S2 all even numbers are below odd numbers.
- Write a Subprogram to copy the even numbers from S1 to S2. The content of S1 after execution must be identical to that before execution. The even numbers in S2 must be in the order they appear in S1.