

DW Series N 03 Stacks and Queues

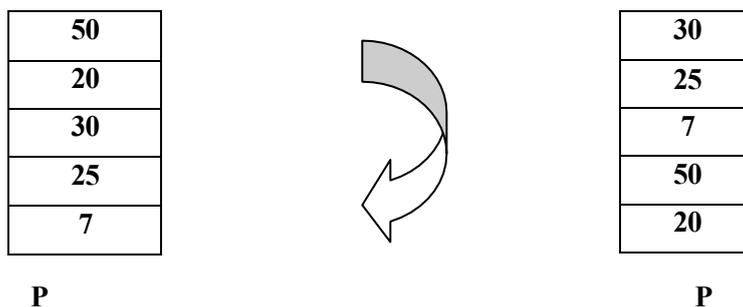
Exercise 01: give the declaration of a dynamic stack of integers and write algorithmically the operations allowing us to create a stack of N integers, to display its even elements and to calculate the sum of its elements. Use directly stack primitive operations (is_empty, Push, Pop, Top, Size).

Exercise 02: Repeat exercise 1 for a dynamic queue of integers. Use directly queue primitive operations (is_empty, Enqueue, Dequeue, Head, Size).

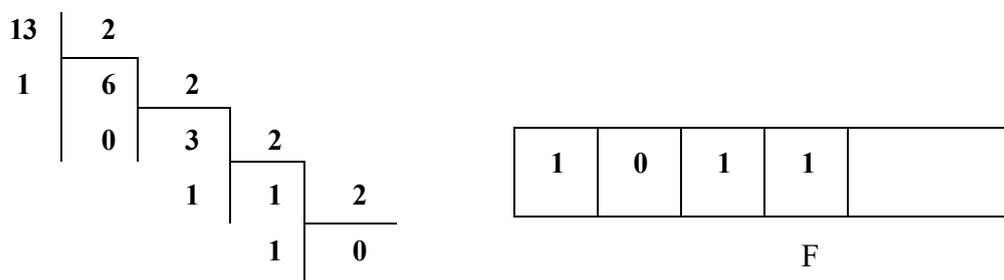
Exercise 03: Using a stack and its primitives, write the function copy (L) that allows creating a copy of the linked list L.

Exercise 04 (circular permutations): write a function `permutation_circ` which receives as an argument a stack P and an integer N ($N \leq \text{Size}(P)$) and performs N successive circular permutations on the stack.

Example with N=2:



Exercise 05: in the example below, row F contains the binary representation of the number 13 (1101).



1. *binary* procedure that receives a decimal number x and creates a queue F containing its binary representation.
2. *decimal* function that receives a queue F containing the binary representation of an integer x and returns its decimal representation.

Exercise 06: We want to implement a program to calculate the approximate printing time of a set of documents threaded into a printer queue. This queue is implemented by a linked list (dynamic queue) where each link contains the name of a document with the number of pages to print.

- 1) Define the data types needed for the implementation of this queue.
- 2) Write the function that returns the number of documents exist in the queue.
- 3) If the time to print one page is 5 seconds, write the procedure that displays the time required to print all documents.

Exercise 07: additional

Using a stack, write an iterative version for the recursive function `power(x, n)` respecting the algorithm of its execution by programming languages.

```
int powerful(int x, int n)
{ if (n==)
    return 1;
else
    return x * power(x, n-1);
}
```

Exercise 08: additional

Repeat exercise 6, considering that each document in the queue is represented by a document name and a queue of pages to print.

Exercise 9: additional

Using a stack, write the function to evaluate a post-fixed arithmetic expression, represented by a linked list, containing only digits (0,1,...9) as operands and the following operators: addition '+', subtraction '-', multiplication '*', and division '/'.

Example: 2 3 * 8 +

