



## Exercicis de recursivitat

Creeu el projecte ***RecursivityExercices*** a IntelliJ, amb el *package* `recursivity`. A dins, hi anireu creant tants fitxers `.java` com exercicis implementeu.

Cal tenir en compte que el codi de tots els exercicis ha d'estar dissenyat modularment, és a dir, amb funcions, aplicant els conceptes explicats a classe. A més a més, cal que els penseu utilitzant el paradigma recursiu.

## Crides simples

### Exercici 1.

Creeu un programa que escrigui els  $N$  primers nombres enters en un fitxer. El fitxer `files/integers.txt` contindrà un llistat de valors  $N$  i el resultat del còmput s'haurà d'escriure al fitxer `results/integers.txt`

### Exercici 2.

Creeu un programa que calculi les multiplicacions especificades en el fitxer `files/multiplications.txt`. Els resultats els emmagatzemareu en un nou fitxer amb ruta relativa `results/multiplications.txt`

### Exercici 3.

Creeu un programa que calculi les exponenciacions del fitxer `files/exp.txt`. Els resultats els emmagatzemareu en un nou fitxer amb ruta relativa `results/exp.txt`

### Exercici 4.

Creeu un programa que calculi el factorial dels nombres que trobareu en el fitxer `files/factorials.txt`. Els resultats els emmagatzemareu en un nou fitxer amb ruta relativa `results/factorials.txt`.

Analitzeu què passa si volem calcular el factorial del número 14.

[CallStack de Java](#)

[Recordatori de la mida dels tipus bàsics de JAVA](#)

### Exercici 5.

Creeu un programa que capgiri els valors d'un array. Al fitxer `files/invert.txt` hi trobareu un llistat d'arrays. Els resultats els emmagatzemareu en un nou fitxer amb ruta relativa `results/invert.txt`

## Crides dobles

### Exercici 1.

Creeu un programa que calculi els  $N$  primers termes de la sèrie Fibonacci. El fitxer `files/fibonacci.txt` conté el llistat de termes a calcular. Els resultats els emmagatzemareu en un nou fitxer amb ruta relativa `results/fibonacci.txt`

## Backtracking

### Exercici 1.

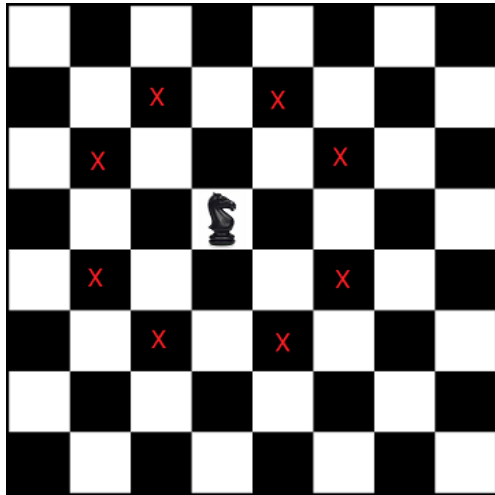
Creeu un programa que, donat un tauler d'escacs, calculi la seqüència de moviments del cavall de tal manera que aquesta figura visiti totes les cel·les del tauler 1 sola vegada. Un cop resolt el problema, escriviu la seqüència obtinguda al fitxer `results/knight.txt`, tal com mostra la Figura **1b**

Dades a tenir en compte:

1. El taule d'escac és de 8x8



2. Donada una cel·la on hi ha un cavall, la Figura 1a mostra tots els moviments possibles (en forma de 'X')



(a) Tots els possibles moviments del cavall

1	50	45	62	31	18	9	64
46	61	32	49	10	63	30	17
51	2	47	44	33	28	19	8
60	35	42	27	48	11	16	29
41	52	3	34	43	24	7	20
36	59	38	55	26	21	12	15
53	40	57	4	23	14	25	6
58	37	54	39	56	5	22	13

(b) Possible seqüència dels moviments del cavall

Figura 1: Imatges d'exemple per al moviment del cavall

### Exercici 2.

Creeu un programa implementi el problema de la motxilla (*knapsack problem*). Aquest problema defineix una motxilla que pot suportar X kg i un conjunt d'objectes amb un pes i un valor monetari determinats. La solució consisteix en trobar tots aquells objectes que, omplint al màxim la motxilla (sense passar-se del pes que pot suportar), sumin el màxim valor monetari possible.

El fitxer `files/knapsack.txt` conté les dades de la motxilla i dels objectes i, un cop resolt el problema, heu d'escriure'n el resultat al fitxer `results/knapsack.txt`

## Divide & Conquer

### Exercici 1.

Creeu un programa que implementi la cerca binària utilitzant recursivitat. Al fitxer `files/binary_search.txt` hi trobareu els elements que componen l'array ordenat i el llistat de valors que es volen buscar. Per cada valor, en realitzareu la cerca i escriureu el resultat obtingut al fitxer `results/binary_search.txt`:

- Si no heu trobat el valor, afegireu al fitxer la línia següent: `valor_X: no trobat`
- Si heu trobat el valor, afegireu al fitxer la línia següent: `valor_X: trobat a la posicióY`

### Exercici 2.

Creeu un programa que implementi l'algorisme *Quicksort* per ordenar un array utilitzant recursivitat. Al fitxer `files/quicksort.txt` hi trobareu un llistat d'arrays. Els heu d'ordenar tots, un a un i escriure'n el resultat obtingut al fitxer `results/quicksort.txt`.

La Figura 2 mostra, esquemàticament, com funciona el *Quicksort*.

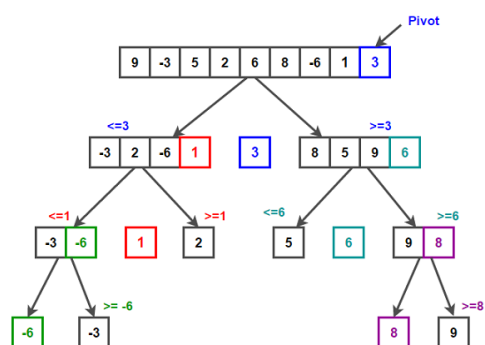


Figura 2: Funcionament de l'algorisme d'ordenació *Quicksort*