

# Exercici 1

Heu de calcular la enèsima potència d'un número fent una calculadora. Creeu una classe **MyCalculator** amb un únic mètode **long power(int, int)**. Rep dos enters, **n** i **p**, com a paràmetres i calcula  **$n^p$** . Si **n** o **p** són negatius, el mètode ha de tornar una excepció que digui «**Ni n ni p poden ser negatius.**». Si tots dos són 0, llavors ha de tornar una excepció que digui «**Ni n ni p poden ser zero.**».

## Format d'entrada

Haureu de llegir des del teclat. Això ho fareu des de la classe principal on hi hagi el main()

Cada línia de l'entrada conté dos enters, **n** i **p**, que són els que es passaran al mètode.

## Condicions

- $0 < n \leq 10$
- $0 < p \leq 10$
- Cada element és un *32-bit integer*.

## Format de sortida

Cada cop que entreu una parella **n** i **p**, es printarà per la consola el resultat de  **$n^p$** . Si **n** o **p** són negatius, la sortida dirà «**Ni n ni p poden ser negatius.**». Si tots dos són 0, la sortida dirà «**Ni n ni p poden ser zero.**».

## Input d'exemple

```
3 5
2 10
0 0
-1 -5
-6 2
```

## Output d'exemple

```
243
1024
Ni n ni p poden ser zero.
Ni n ni p poden ser negatius.
Ni n ni p poden ser negatius.
```

## Explicació

En els dos primers casos **n** i **p** són positius tots dos. Es calcula la potència sense problemes.

En el tercer cas tots dos són 0, per tant es printa el missatge de l'excepció.

En els dos últims casos com a mínim un número és negatiu, per tant es printa el missatge de l'excepció.

## Exercici 2

Heu de calcular la divisió de dos enters,  $x$  i  $y$ . Si  $x$  o  $y$  no són **32 bits signed integers** o  $y$  és **0**, saltarà una excepció que haureu de gestionar.

### Format d'entrada

Haureu de llegir des del teclat.

Primer entrareu un enter i després un altre.

### Input d'exemple

```
12
4
```

### Output d'exemple

```
3
```

### Input d'exemple

```
12
holaaaaa!!
```

### Output d'exemple

```
java.util.InputMismatchException
```

### Input d'exemple

```
12
0
```

### Output d'exemple

```
java.lang.ArithmeticException: / by zero
```

### Input d'exemple

```
12,452
0
```

### Output d'exemple

```
java.util.InputMismatchException
```

## Exercici 3

En aquesta activitat heu de realitzar una classe que calculi la solució d'una equació de segon grau.

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

A l'entrada, pel teclat entrareu els termes **a**, **b** i **c** de l'equació, que seran números enters. Haureu de comprovar que s'entren correctament números enters i que el discriminant no és menor que 0:

### Discriminant Notes

$$D = b^2 - 4ac$$

$$D > 0 \quad 2 \text{ real roots}$$

$$D = 0 \quad 1 \text{ real root}$$

$$D < 0 \quad 2 \text{ complex roots}$$

En cas que el discriminant sigui menor que 0 haureu de llençar una excepció que haureu de crear vosaltres.

A la classe principal haureu de capturar les excepcions i gestionar-les.

### Format d'entrada

Haureu de llegir des del teclat.

A cada línia entrareu un enter.

### Condicions

- El discriminant no pot ser negatiu
- Cada element és un *32-bit integer*.
- **a** no pot ser 0.

**Format de sortida**

Cada línia printarà la solució de l'equació o haurà d'informar de la excepció que ha impossibilitat el càlcul.