**JAVA - Basics**

**Atelier TP N° 5**

Exceptions

**Exercice 1 :**

Donner une classe d'entiers naturels (positifs ou nuls) et disposant :

• D’un constructeur à un argument de type ***int***; il générera une exception ***ErrConst***si la valeur de son argument est négative.

• De méthodes statiques de somme, de différence et de produit de deux naturels ; elles généreront respectivement des exceptions ***ErrSom,*** ***ErrDiff***et ***ErrProd***lorsque le résultat ne sera pas représentable ; la limite des valeurs des naturels sera fixée à la plus grande valeur du type ***int.***

• Une méthode d’accès ***getN()***fournissant sous forme d’un ***int***la valeur de l’entier naturel.

Toutes les classes exception dérivent d’une classe ***ErrNat.***

Écrire deux exemples d’utilisation de la classe :

• L’un se contentant d’intercepter sans discernement les exceptions de type dérivé de

***ErrNat,***

• L’autre qui explicite la nature de l’exception en affichant les informations disponibles.

Les deux exemples pourront figurer dans deux blocs ***try***d’un même programme.

**Exercice 2 :**

Dans cet exercice on va créer la classe Bouteille qui permet d’instancier des bouteilles d’eau minérale de deux tailles différentes :

* 1 : 50 cl.
* 2 : 75 cl.

1 - Implémenter la classe Bouteille (attributs, getters…).

2 – Gérer les erreurs de construction à l’aide d’une classe exception appropriée.

On voudrait emballer ces bouteilles dans des cartons selon la taille :

* Pour bouteille taille 1 : carton taille 1 prend 12 bouteilles.
* Pour bouteille taille 2 : carton taille 2 prend 8 bouteilles.

3 – Implémenter la classe Carton (dérivée en 2 classes filles) et la méthode ajouter qui permet remplir un carton.

4 – Gérer les exceptions dues à l'ajout dans un carton : Mauvais choix de bouteille et dépassement de capacité d’un carton.

**Exercice 3 :**

Créer une classe qui permet d'instancier des équations au second degré. On dispose d'un constructeur 3 paramètres et de getters.

Ajouter la méthode affichEqua() pour afficher l'équation sous la forme ***''ax² + bx +c''.***

Donner la méthode void solution() qui calcule et affiche les solution dans **R**.

Gérer les exceptions possibles : pas de solution et solution dans l'ensemble **C**.