Furno Tristan
Samuel Le Berre
Groupe A
M2105

03/04/2017

# Partie A

## Question 1

```
/**
* Classe tableau TP2 IHM
* @author Tristan Furno - Samuel LeBerre
*/
public class Tableau{
  /**
  * Longueur du Tableau
  */
  private int longueur;

  /**
  * tableau d'entiers
  */
  private int[] valeurs;

  /**
  * Constructeur de la classe tableau
  * @param nbCases nombres de valeurs dans le tableau
  */
  public Tableau(int nbCases){
      this.longueur = nbCases;
      this.valeurs = new int[nbCases];
  }

  /**
  * Getter de la longueur
  * @return longueur du tableau
  */
  public int getLongueur(){return this.longueur;}

  /**
  * Retourne la valeur choisie en index
  * @return la valeur de l'index choisie
  * @param index - l'index a retourner
  */
  public int getValeur(int index){
    return this.valeurs[index];
  }

  /**
  * Methode pour set une valeur du tableau
  * @param val - valeur a entrer dans le tableau
  * @param index - index au quel la valeur sera rentree
  */
  public void setValeur(int val, int index){
      this.valeurs[index] = val;
  }
}
```

### Question 2

```
Case d'indice négatif + case d'indice supérieur à la taille du tableau (erreur à l'exécution) :
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: -1
        at Tableau.getValeur(Tableau.java:37)
        at TestTableau.main(TestTableau.java:24)

Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 11
        at Tableau.getValeur(Tableau.java:37)
        at TestTableau.main(TestTableau.java:26)

Case sans avoir initialisé le tableau (erreur à la compilation) :
..\src\TestTableau.java:28: error: variable tab2 might not have been initialized
                System.out.println(tab2.getValeur(12));
                                   ^
1 error
```

### Question 3

Lorsque l'erreur se trouve à l'exécution la JVM renvoie une exception indiquant dans quelles méthodes l'erreur a eu lieu et où l'erreur renvoie : dans la question 2 l'erreur se situe dans la méthode main de la classe TestTableau à la ligne 24, la ligne 24 appelle la méthode getValeur de la classe Tableau à la ligne 37, lieu où est créée l'erreur.

## Partie B

### Question 1

```
public Tableau(String fileName){
    FileReader file = new FileReader(fileName);
    BufferedReader br = new BufferedReader(file);
    this.longueur = Integer.parseInt(br.readLine());
    this.valeurs = new int[this.longueur];
    for(int i = 0; i < this.longueur ; i++){
      this.valeurs[i] = Integer.parseInt(br.readLine());
    }
    br.close();
  }
```

### Question 2

```
Java n'autorise pas la compilation.

..\src\Tableau.java:27: error: unreported exception FileNotFoundException; must be caught or declared to be thrown
    FileReader file = new FileReader(fileName);
                      ^
..\src\Tableau.java:29: error: unreported exception IOException; must be caught or declared to be thrown
    this.longueur = Integer.parseInt(br.readLine());
                                                 ^
..\src\Tableau.java:31: error: unreported exception IOException; must be caught or declared to be thrown
      this.valeurs[i] = Integer.parseInt(br.readLine());
                                                   ^
..\src\Tableau.java:33: error: unreported exception IOException; must be caught or declared to be thrown
```

```
    br.close();
            ^
4 errors
```

Le nouveau constructeur ne gère pas les IOExceptions qui pourraient survenir dans le code. Toutes les exceptions (sauf java.lang) doivent être attrapées à l'aide d'un try catch.

## Question 3

```
public Tableau(String fileName){
    FileReader file;
    BufferedReader br;
    try{
    file = new FileReader(fileName);
    br = new BufferedReader(file);
    this.longueur = Integer.parseInt(br.readLine());
    this.valeurs = new int[this.longueur];
    for(int i = 0; i < this.longueur ; i++){
      this.valeurs[i] = Integer.parseInt(br.readLine());
    }
    br.close();
    }
    catch(Exception e){
      e.printStackTrace();
    }
  }
```

Le constructeur de FileReader demande que l'exception FileNotFoundException soit gérée.
Les méthodes readLine() et close() de la classe BufferedReader demande que l'exception IOException soit gérée.

## Question 4

Le programme de test avec le nouveau constructeur compile correctement. Mais à l'exécution il affiche toutes les méthodes qui cherchent le fichier à lire.

```
java.io.FileNotFoundException: toto.txt (Le fichier spécifié est introuvable)
        at java.io.FileInputStream.open0(Native Method)
        at java.io.FileInputStream.open(Unknown Source)
        at java.io.FileInputStream.<init>(Unknown Source)
        at java.io.FileInputStream.<init>(Unknown Source)
        at java.io.FileReader.<init>(Unknown Source)
        at Tableau.<init>(Tableau.java:30)
        at TestTableau.main(TestTableau.java:8)
```

## Question 5

```
  public Tableau(String fileName) throws Exception{
    FileReader file;
    BufferedReader br;
    file = new FileReader(fileName);
    br = new BufferedReader(file);
    this.longueur = Integer.parseInt(br.readLine());
    this.valeurs = new int[this.longueur];
    for(int i = 0; i < this.longueur ; i++){
      this.valeurs[i] = Integer.parseInt(br.readLine());
    }
    br.close();
  }
```

```
public class TestTableau{

	public static void main(String[] args){

		try{
			Tableau tab1 = new Tableau("toto.txt");
		}
		catch(Exception e){
			e.printStackTrace();
		}
}
```

On enlève le try catch du constructeur, il faut donc le rajouter dans les méthodes qui utilisent le constructeur (autour du constructeur).

## Question 6

```
import java.util.Scanner;
 import java.io.*;
 import java.lang.*;
public class TestTableau{

	public static void main(String[] args){

		try{
		Tableau tab1 = new Tableau("toto.txt");
		}
		catch(NumberFormatException e){
			System.out.println("Fichier vide");
			//System.out.println("Votre fichier n a pas ete cree : recreez un tableau");
		}
		catch(IOException e){
			System.out.println("Fichier inexistant");
		}
		Tableau tab2;



	public static void main(String[] args){
		String fichier = "toto.txt";
		Tableau tab1 = null;
		while(tab1 == null){
			try{
				tab1 = new Tableau(fichier);
			}
			catch(NumberFormatException e){
				Scanner sc = new Scanner(System.in);
				System.out.println("Veuillez saisir un nouveau nom de fichier non vide");
				fichier = sc.nextLine();
			}
			catch(IOException e){
				Scanner sc = new Scanner(System.in);
				System.out.println("Veuillez saisir un nouveau nom de fichier existant");
				fichier = sc.nextLine();
			}
	}
```

### Question 7

Les exceptions runtime n'ont pas besoin d'être mises dans le throw mais sont quand même utilisées. Alors que les exceptions vues dans l'exercice 2 doivent être déclarées derrière un throw et gérées dans la méthode appelante.

## Partie C

### Question 1

```
public int getValeur(int index) throws Exception{
    int ret = 0;
    if(index < 0 ){
      throw new Exception("L'indice ne se trouve pas dans le tableau");
    }else if(index > this.longueur-1){
      throw new Exception("L'indice ne se trouve pas dans le tableau");
    }else{
      ret = this.valeurs[index];
    }
    return ret;
  }

  public void setValeur(int val, int index) throws Exception{
    if(index < 0 ){
      throw new Exception("L'indice ne se trouve pas dans le tableau");
    }else if(index > this.longueur-1){
      throw new Exception("L'indice ne se trouve pas dans le tableau");
    }else{
      this.valeurs[index] = val;
    }
  }
```

### Question 2

```
import java.io.*;
public class TableauException extends Exception{

  public TableauException(){
    super("L'indice ne se trouve pas dans le tableau");
  }
}
```

On crée une classe exception qui hérite de la classe exception et lui passe un message par son paramètre de constructeur. On modifie les :
`throw new Exception("L'indice ne se trouve pas dans le tableau");`
Par des :
`throw new TableauExcetion();`
Qui renvoie la chaine string :
`L'indice ne se trouve pas dans le tableau`

```
D:\Cours\IHM\TP2\Travail en cours\ws>java TestTableau tata.txt
L'indice ne se trouve pas dans le tableau
```

### Question 3

Nous avions déjà réalisé cette tâche lors de la question 6 de la partie B.