Algorithmes et Pensée Computationnelle

Fonctions, mémoire et exceptions

Le but de cette séance est d'approfondir vos connaissances en programmation. Au terme de cette séance, l'étudiant sera capable de :

- définir une fonction et l'utiliser dans un programme,
- utiliser des librairies contenant des fonctions prédéfinies,
- connaître quelle est la portée d'une variable,
- comprendre comment fonctionne une pile d'exécution (call stack)

1 **Gestion des exceptions**

Question 1: (**0** 5 minutes) Qu'affiche le programme suivant? **Types d'erreurs (Python)**

```
value1 = "Algorithms"
 2
     value2 = 4
 3
 4
     try:
 5
       size = len(value1)
       result = size/value2
 6
 7
       print(f"Le résultat de la division est: {result}")
 8
 9
     except Exception as error:
10
       print("On ne peut pas effectuer l'opération")
11
12
     try:
13
       result = value 1/2
14
       print(f''Le résultat de la division est: {result}'')
15
16
     except TypeError as error:
       print("On ne peut pas diviser une chaîne de caractères")
17
```

Conseil

Utilisée sur une chaîne de caractères, la fonction len() renvoie le nombre de caractères de la chaîne.

>_ Solution

Le résultat de la division est: 2.5

On ne peut pas diviser une chaîne de caractères.

Question 2: (**0** 5 minutes) Qu'affiche le programme suivant? **Types d'erreurs (Python)**

```
1
    value1 = 4
    value2 = ""
2
3
5
      count = len(value2)
6
      result= value1/count
   except ZeroDivisionError as error:
      print("Nous ne pouvons pas diviser un nombre par 0")
```

Conseil

La chaîne de caractères vide est représentée par ""

>_ Solution

Nous ne pouvons pas diviser un nombre par 0.

Question 3: (5 minutes) Qu'affiche le programme suivant? Types d'erreurs (Python)

```
value1 = "Algorithms"
 1
 2
     value2 = 4
 3
 4
 5
     try:
 6
       decimal = float(value2)
 7
     except ValueError as error:
 8
       print("Nous ne pouvons pas convertir un entier en décimal")
 9
10
     finally:
11
       try:
         value2 = int(value1)
12
13
       except ValueError as error:
14
         print("Nous ne pouvons pas convertir une chaîne de caractères en nombre")
```

Conseil

- La fonction float() permet de convertir une variable en nombre à décimal.
- La fonction int() permet de convertir une variable en nombre entier.

>_ Solution

Nous ne pouvons pas convertir une chaîne de caractères.

Question 4: (*5 minutes*) **Gestion d'erreurs** (**Java**) Le programme suivant est incorrect. Que devezvous modifier pour qu'il fonctionne correctement?

```
public class Question8 {
 3
       public static void division(int a, int b) throws ArithmeticException{
 4
 5
            throw new ArithmeticException();
 6
          }else{
 7
            float result = a/b;
            System.out.println("Le résultat de la division de " + a + "/" + b + "= " + result);
 8
 9
10
     }
       public static void main(String args[]){
11
12
          int value1 = 2;
13
         int value2 = 4;
14
         try {
15
            division(value1,value2);
16
            value2 = 0;
            division(value1,value2);
17
          }catch(IndexOutOfBoundsException err){
18
19
            System.out.println("Nous ne pouvons pas effectuer une division par 0");
20
21
       }
     }
```

Conseil

Référez-vous à la diapositive 26 du cours de cette semaine.

>_ Solution

À l'intérieur de la fonction division, lorsque b est égal à 0, le programme lève une exception de type ArithmeticException. Mais dans le main, l'erreur qui est interceptée est IndexOutOfBoundsException. Pour corriger cette erreur, une des solutions est de remplacer IndexOutOfBoundsException par ArithmeticException.