CS159

Nathan Sprague

What’s wrong with the following code?

```java
/***************************************************
* Return the mean, or -1 if the array has length 0.
****************************************************/
public static double mean(double[] numbers)
      {
         double sum = 0;
         double result;

         if (numbers == null || numbers.length == 0)
         {
            result = -1;
         }
         else
         {
            for (int i = 0; i < numbers.length; i++)
            {
               sum += numbers[i];
            }
            result = sum / numbers.length;
         }
         return result;
      }
```

Why Exceptions?

- Sometimes there is no appropriate return value that can be used to indicate an error has occurred. (Let's use exceptions to improve this code...)

Why Exceptions?

- Sometimes there is no appropriate return value that can be used to indicate an error has occurred. (Let's use exceptions to improve this code...)
- Exceptions provide flexibility in deciding where a particular problem should be handled. If an exception occurs, a method may:
  - "Pass the buck" by using the throws keyword. The exception will be handled somewhere higher-up in the call stack.
  - Deal with the exception using a try/catch block.

Fixed Mean Calculator

```java
    public static double mean(double[] numbers)
    {
        double sum = 0;
        double result;

        if (numbers == null || numbers.length == 0)
        {
            throw new IllegalArgumentException("Invalid array.");
        }

        for (int i = 0; i < numbers.length; i++)
        {
            sum += numbers[i];
        }
        result = sum / numbers.length;

        return result;
    }
```

Example...

Let's write a Java program that reads a text file containing simple mathematical expressions, and writes the results:

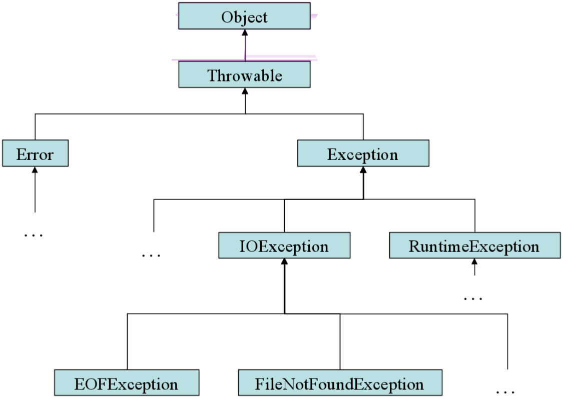
- If the input file contains:

```
2 + 3
22 / 2
3 * 2
```

- Terminal output should be:

```
5
11
6
```

Exception Class Hierarchy



What will be printed?

```
fileName = "NONEXISTENTFILE.txt";
System.out.print("A ");
try
{
   System.out.print("B ");
   file = new File(fileName);
   scanner = new Scanner(file);
   System.out.print("C ");

}
catch (FileNotFoundException e)
{
   System.out.print("D ");
}
finally
{
   System.out.print("E ");
}
System.out.print("F ");
```

Question

Characterize the following code, assuming `numbers` is an array of doubles. (There are no syntax errors.)

```java
double sum = 0;
try
{
    for (int i = 0; i <= numbers.length; i++)
    {
        sum += numbers[i];
    }
}
catch (ArrayIndexOutOfBoundsException e)
{
    //Do nothing.
}
System.out.println(sum);
```

1. Correct result, appropriate use of exception handling.
2. Incorrect result, appropriate use of exception handling.
3. Correct result, inappropriate use of exception handling
4. Incorrect result, inappropriate use of exception handling.

Question

Characterize the following code, assuming `numbers` is an array of doubles. (There are no syntax errors.)

```java
double sum = 0;
try
{
   for (int i = 0; i <= numbers.length; i++)
   {
      sum += numbers[i];
   }
}
catch (ArrayIndexOutOfBoundsException e)
{
   //Do nothing.
}
System.out.println(sum);
```

1. Correct result, appropriate use of exception handling.
2. Incorrect result, appropriate use of exception handling.
3. Correct result, inappropriate use of exception handling
4. Incorrect result, inappropriate use of exception handling.

3.

Handling vs. Throwing

The specification for the following method indicates that it should "throw an exception of type IllegalArgumentException if amount is less than 0." Is this implementation correct?

```java
    public void increaseValue(int amount)
     {
         try
         {
             if (amount < 0)
             {
                 throw new IllegalArgumentException();
             }
             value += amount;
         }
         catch (IllegalArgumentException e)
         {
             System.out.println("Negative amount not allowed!");
         }
     }
```

Handling vs. Throwing

The specification for the following method indicates that it should “throw an exception of type IllegalArgumentException if amount is less than 0.” Is this implementation correct?

```java
    public void increaseValue(int amount)
     {
         try
         {
             if (amount < 0)
             {
                 throw new IllegalArgumentException();
             }
             value += amount;
         }
         catch (IllegalArgumentException e)
         {
             System.out.println("Negative amount not allowed!");
         }
     }
```

No.