

Error Handling

Bài 1 (2.5 điểm)

Cho một chương trình Calculator để thực hiện các tác vụ tính toán (+, -, *, /) giữa hai số nguyên dưới đây. Để thực hiện việc tính toán, người dùng sẽ nhập vào cho chương trình một biểu thức từ dòng lệnh có dạng [số nguyên] <phép toán> [số nguyên]. Ví dụ:

```
Nhập vào một biểu thức: 2 + 3
Kết quả: 2 + 3 = 5
Nhập vào một biểu thức: 17 - 10
Kết quả: 17 - 10 = 7
Nhập vào một biểu thức: 4 * 5
Kết quả: 4 * 5 = 20
Nhập vào một biểu thức: 9 / 3
Kết quả: 9 / 3 = 3
Nhập vào một biểu thức:
```

Chương trình sẽ tính biểu thức được tạo bởi hai số nguyên đó và in kết quả tìm được ra màn hình. Dưới đây là đoạn code của chương trình Calculator:

```
package lab_assignment9;

import java.util.*;
public class Calculator {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        String[] arguments;
        do {
            System.out.print("Nhập vào một biểu thức: ");
            String expression = input.nextLine(); // Nhập vào chuỗi biểu thức
            expression = expression.trim(); // Chuẩn hóa chuỗi
            arguments = expression.split(" "); // Cắt chuỗi dựa vào kí tự
            khoảng trắng và lưu từng thành phần vào một mảng

            if (arguments.length != 3) {
                System.out.println("Chuỗi nhập vào phải có dạng [toán hạng]
<phép toán> [toán hạng]");

                System.out.println("=====
=====");
            }

        } while (arguments.length != 3);

        // Kết quả của biểu thức
        int result = 0;

        switch (arguments[1].charAt(0)) {
            case '+': result = Integer.parseInt(arguments[0]) +
Integer.parseInt(arguments[2]); break;
```

```
        case '-': result = Integer.parseInt(arguments[0]) -  
Integer.parseInt(arguments[2]); break;  
        case '*': result = Integer.parseInt(arguments[0]) *  
Integer.parseInt(arguments[2]); break;  
        case '/': result = Integer.parseInt(arguments[0]) /  
Integer.parseInt(arguments[2]); break;  
    }  
  
    // Xuất kết quả ra màn hình  
    System.out.println("Kết quả: " + arguments[0] + " " + arguments[1] + " "  
+ arguments[2] + " = " + result);  
}
```

Tuy nhiên, chương trình này sẽ bị lỗi nếu

- + Các toán hạng nhập vào không phải là số nguyên
- + Biểu thức thực hiện phép chia cho 0
- + Các toán tử không phải là (+, -, *, /)

Yêu cầu: hãy xây dựng các trình xử lý biệt lệ để ngăn ngừa và thông báo cho người dùng mỗi khi có 1 trong các lỗi trên xảy ra.

Đáp án

```
package lab_assignment9;  
  
import java.util.*;  
public class Calculator {  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);  
        String[] arguments;  
        do {  
            System.out.print("Nhập vào một biểu thức: ");  
            String expression = input.nextLine(); // Nhập vào chuỗi biểu thức  
            expression = expression.trim(); // Chuẩn hóa chuỗi  
            arguments = expression.split(" "); // Cắt chuỗi dựa vào kí tự  
            khoảng trắng và lưu từng thành phần vào một mảng  
  
            if (arguments.length != 3) {  
                System.out.println("Chuỗi nhập vào phải có dạng [toán hạng]  
<phép toán> [toán hạng]");  
  
                System.out.println("=====  
====");  
            }  
  
        } while (arguments.length != 3);  
  
        try {  
            // Kết quả của biểu thức  
            int result = 0;
```

```
        switch (arguments[1].charAt(0)) {
            case '+': result = Integer.parseInt(arguments[0]) +
Integer.parseInt(arguments[2]); break;
            case '-': result = Integer.parseInt(arguments[0]) -
Integer.parseInt(arguments[2]); break;
            case '*': result = Integer.parseInt(arguments[0]) *
Integer.parseInt(arguments[2]); break;
            case '/': result = Integer.parseInt(arguments[0]) /
Integer.parseInt(arguments[2]); break;
            default: throw new Exception("Lỗi: toán tử " + arguments[1]
+ " không nằm trong (+, -, *, /)");
        }

        // Xuất kết quả ra màn hình
        System.out.println("Kết quả: " + arguments[0] + " " +
arguments[1] + " " + arguments[2] + " = " + result);
    }
    catch(NumberFormatException Ex1) {
        System.out.println("Lỗi: số hạng nhập vào không phải là số
nguyên!");
    }

    catch(ArithmeticException Ex2) {
        System.out.println("Lỗi: bạn đang thực hiện phép chia cho 0!");
    }

    catch(Exception Ex3) {
        System.out.println(Ex3.getMessage());
    }
}
}
```

Bài 2 (2.5 điểm)

Viết một chương trình đáp ứng các yêu cầu sau:

- Tạo ra một mảng có 5 phần tử số nguyên ngẫu nhiên (random)
- Yêu cầu người dùng nhập vào chỉ số của một phần tử trong mảng và xuất giá trị của phần tử đó ra màn hình. Tuy nhiên, nếu chỉ số của phần tử nằm ngoài phạm vi biểu diễn của mảng thì hãy xuất thông báo ra màn hình “Chỉ số phần tử vượt quá kích thước mảng”

Đáp án

```
package lab_assignment9;

import java.util.Scanner;

public class Bai2 {
    public static void main(String[] args) {
        int[] a = {1, 2, 3, 4, 5};
        Scanner sc = new Scanner(System.in);
        System.out.print("Hãy nhập vào chỉ số một phần tử trong mảng (length = 5) ");
        int index = sc.nextInt();

        try {
            System.out.println("Giá trị của phần tử thứ " + index + " là " + a[index]);
        } catch (ArrayIndexOutOfBoundsException ex) {
            System.out.println("Chỉ số phần tử vượt quá kích thước của mảng!");
        }
    }
}
```

Bài 3 (2.5 điểm)

Cho một lớp hình tam giác (Triangle) dưới đây được biểu diễn bởi độ dài 3 cạnh. Để biết hình tam giác này có hợp lệ hay không thì ta phải kiểm tra các độ dài cạnh theo qui tắc: tổng độ dài của hai cạnh bất kì phải lớn hơn độ dài của cạnh còn lại. Yêu cầu: hãy xây dựng thêm một lớp biệt lệ có tên là `IllegalTriangleException` để kiểm tra tính hợp lệ của tam giác, và hãy sử dụng biệt lệ này trong phương thức khởi tạo của `Triangle` để kiểm tra các tham số về độ dài cạnh truyền vào.

```
package lab_assignment9;

class Triangle {
    int a; // Độ dài cạnh thứ 1
    int b; // Độ dài cạnh thứ 2
    int c; // Độ dài cạnh thứ 3

    // Phương thức khởi tạo
    public Triangle(int a, int b, int c) {
        this.a = a;
        this.b = b;
        this.c = c;
    }

    // Tính chu vi
    public int calcPerimeter() {
        return (this.a + this.b + this.c);
    }

    // Tính diện tích
    public double calcArea() {
        // Công thức Heron tính diện tích tam giác dựa vào nửa chu vi
        double p = this.calcPerimeter() / 2;
        return Math.sqrt(p * (p - a) * (p - b) * (p - c));
    }

    public static void main() {
        Triangle triangle = new Triangle(3, 3, 6);
        System.out.println("Chu vi hình tam giác: " + triangle.calcPerimeter());

        System.out.println("Diện tích hình tam giác: " + triangle.calcArea());
    }
}
```

Đáp án

```
package lab_assignment9;

public class Triangle {
    int a; // Độ dài cạnh thứ 1
    int b; // Độ dài cạnh thứ 2
    int c; // Độ dài cạnh thứ 3

    // Phương thức khởi tạo
    public Triangle(int a, int b, int c) throws IllegalTriangleException {
```

```
        if (a + b <= c || a + c <= b || b + c <= a)
            throw new IllegalArgumentException(a, b, c);

        this.a = a;
        this.b = b;
        this.c = c;
    }

    // Tính chu vi
    public int calcPerimeter() {
        return (this.a + this.b + this.c);
    }

    // Tính diện tích
    public double calcArea() {
        // Công thức Heron tính diện tích tam giác dựa vào nửa chu vi
        double p = this.calcPerimeter() / 2;
        return Math.sqrt(p * (p - a) * (p - b) * (p - c));
    }

    public static void main(String[] args) {
        try {
            Triangle triangle = new Triangle(3, 4, 5);
            System.out.println("Chu vi hình tam giác: " +
triangle.calcPerimeter());
            System.out.println("Diện tích hình tam giác: " +
triangle.calcArea());
        } catch (IllegalArgumentException Ex) {
            System.out.println(Ex.getMessage());
        }
    }
}

class IllegalArgumentException extends Exception{

    public IllegalArgumentException(int a, int b, int c) {
        super("Số đo các cạnh (" + a + ", " + b + ", " + c + ") của tam giác không
hợp lệ!");
    }
}
}
```

Bài 4 (2.5 điểm)

Cho đoạn chương trình biểu diễn cách chuyển đổi một số từ hệ cơ số 16 sang hệ cơ số 10 dưới đây. Từ chương trình này hãy thêm vào biệt lệ `NumberFormatException` để kiểm tra nếu chuỗi nhập vào không phải là một chuỗi hex (chuỗi biểu diễn số ở hệ cơ số 16)

```
package lab_assignment9;

import java.util.Scanner;

public class Hex2Dec { // Lớp chuyển đổi số ở hệ cơ số 16 sang hệ cơ số 10
    public static void main() {
        Scanner input = new Scanner(System.in);
        System.out.print("Nhập vào một chuỗi hex (số ở hệ cơ số 16): ");
        String hex = input.nextLine();

        System.out.println("Giá trị của " + hex + " ở hệ cơ số 10 là: " +
            hexToDecimal(hex.toUpperCase()));
    }

    public static int hexToDecimal(String hex) {
        int decimalValue = 0;
        for (int i = 0; i < hex.length(); i++) {
            char hexChar = hex.charAt(i);
            decimalValue = decimalValue * 16 + hexCharToDecimal(hexChar);
        }
        return decimalValue;
    }

    public static int hexCharToDecimal(char hexChar) {
        if (hexChar >= 'A' && hexChar <= 'F')
            return 10 + hexChar - 'A';
        else // hexChar là các kí tự '0', '1', ..., '9'
            return hexChar - '0';
    }
}
```

Đáp án

```
package lab_assignment9;

import java.util.Scanner;

public class Hex2Dec { // Lớp chuyển đổi số ở hệ cơ số 16 sang hệ cơ số 10
    public static void main() {
        Scanner input = new Scanner(System.in);
        System.out.print("Nhập vào một chuỗi hex (số ở hệ cơ số 16): ");
        String hex = input.nextLine();
        try {
            System.out.println("Giá trị của " + hex + " ở hệ cơ số 10 là: " +
                hexToDecimal(hex.toUpperCase()));
        } catch (NumberFormatException Ex) {
            System.out.println(Ex.getMessage());
        }
    }
}
```



```
public static int hexToDecimal(String hex) {  
    int decimalValue = 0;  
    for (int i= 0; i<hex.length(); i++) {  
        char hexChar = hex.charAt(i);  
        decimalValue = decimalValue * 16 + hexCharToDecimal(hexChar);  
    }  
    return decimalValue;  
}  
  
public static int hexCharToDecimal(char hexChar) {  
    if (hexChar >= 'A' && hexChar <= 'F')  
        return 10 + hexChar - 'A';  
    else // hexChar là các kí tự '0', '1',..., '9'  
        return hexChar - '0';  
}  
}
```