Errors & Assertions
Chap. 5.4 – 5.6

User Errors

- What is the object of our program?
  - Correctness
  - Efficiency
  - Robustness
- How do we handle input errors?
  - Check data ahead of time

Algorithm for Handling Mismatched Input

Scenario: We ask the user for an integer, and we want to discard everything that isn’t an integer.

prompt user for input
read input from user
while(user’s input is not an integer)
generate error message
prompt for different input
read input from user

- What is wrong with this in Java?
Look ahead at user input...

Scenario: We ask the user for an integer, and we want to discard everything that isn’t an integer.
- prompt user for input
- while (looking at the user’s input, it is not an integer)
- generate error message
- prompt user for input again

• What is wrong here?

Discard Bad Input

Scenario: We ask the user for an integer, and we want to discard everything that isn’t an integer.
- prompt user for input
- while (looking at the user’s input, it is not an integer)
- discard the bad input
- generate error message
- prompt user for input again
- read integer input from user

Discard Bad Input...

```java
int num;
Scanner input = new Scanner(System.in);
System.out.print("Enter an integer: ");
while (!input.hasNextInt()) {
    input.next();
    System.out.println("Come on, I said to enter an integer!!!");
    System.out.print("Enter an integer: ");
}
num = input.nextInt(); //Now, read the integer value
```
**getIntegerInput Method**

```java
public static void main(String[] args) {
    int num;
    String prompt = "Enter an integer value: ";
    num = getIntegerInput(prompt);
}

public static int getIntegerInput(String prompt) {
    Scanner input = new Scanner(System.in);
    System.out.print(prompt);
    while (!input.hasNextInt()) {
        input.next();
        System.out.println("Come on, I said to enter an integer!!!");
        System.out.print(prompt);
    }
    return input.nextInt(); //Now, read the integer value
}
```

---

**How would you handle divide by zero?**

```java
public static void main(String[] args) {
    int num, total_score;
    float average;
    String prompt;

    prompt = "Enter you total test score: ";
    total_score = getIntegerInput(prompt);

    prompt = "Enter the number of tests: ";
    num = getIntegerInput(prompt);

    average = total_score / num;
    System.out.println("Your average test score is: " + average);
}
```

---

**Handling divide by zero...**

```java
public static void main(String[] args) {
    int num, total_score;
    float average;
    String prompt;

    prompt = "Enter you total test score: ";
    total_score = getIntegerInput(prompt);

    prompt = "Enter the number of tests: ";
    num = getIntegerInput(prompt);
    while(num == 0) {
        System.out.println("You cannot have zero tests!");
        num = getIntegerInput(prompt);
    }
    average = total_score / num;
    System.out.println("Your average test score is: " + average);
}
```
Another way to handle errors...

```java
public static int getIntegerInput(String prompt) {
    Scanner input = new Scanner(System.in);
    int value;
    boolean bad_input=true;

    while(bad_input) {
        try {
            System.out.print(prompt);
            value = input.nextInt();
            bad_input=false;
        } catch (Exception ex) {
            input.next(); //Still have to discard bad input
            System.out.println("Come on, I said to enter an integer!!!");
        }
    }
    return value;
}
```