Repetition

CSC 1051 – Data Structures and Algorithms I

Dr. Mary-Angela Papalaskari
Department of Computing Sciences
Villanova University

Course website:
www.csc.villanova.edu/~map/1051/
Topic Thread

- 2.1 Character Strings
- 2.2 Variables, Assignment
- 2.3 Data Types, in particular int, double
- 2.4 Expressions (simple)
- 2.5 Data Conversion
- 2.6 Interactive Programs
- 5.1 Boolean Expressions
- 5.2 The if Statement
- 5.4 The while Statement
Flow of Control

The order of statement execution

• Unless specified otherwise, the order of statement execution through a method is \textit{linear}

• Some programming statements allow us to:
  – decide whether or not to execute a particular statement
  – execute a statement over and over, repetitively

• These decisions are based on \textit{boolean expressions} (or \textit{conditions}) that evaluate to \texttt{true} or \texttt{false}
Flow of Control

The order of statement execution

• Unless specified otherwise, the order of statement execution through a method is *linear*

• Some programming statements allow us to:
  – decide whether or not to execute a particular statement
  – execute a statement over and over, repetitively

• These decisions are based on *boolean expressions* (or *conditions*) that evaluate to *true* or *false*
The while Statement

• A while statement has the following syntax:

```
while ( condition )
  statement;
```

• If the condition is true, the statement is executed

• Then the condition is evaluated again, and if it is still true, the statement is executed again

• The statement is executed repeatedly until the condition becomes false
Logic of a while Loop

- **condition evaluated**
  - **true**
  - **false**

- **statement**
The while Statement

• An example of a while statement:

```java
int count = 1;
while (count <= 3)
{
    System.out.println (count);
    count++;
}
```

• If the condition of a while loop is false initially, the statement is never executed

• Therefore, the body of a while loop will execute zero or more times
the while statement

```java
int count = 1;
while (count <= 3)
{
    System.out.println (count);
    count++;
}
```
The while Statement

```java
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;  
}
```
The while Statement

```java
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;
}
```

count <= 3 is true
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;
}

Output:
1
The while Statement

```java
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;
}
```

Output:

1

The while Statement

```java
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;
}
```

**Output:**
```
1
```
The while Statement

```java
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;
}
```

Output:
1
2
The while Statement

```
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;
}
```

Output:

1
2
3
The while Statement

```java
int count = 1;
while (count <= 3)
{
    System.out.println (count);
    count++;
}
```

Output:

```
1 2
```
int count = 1;
while (count <= 3) {
    System.out.println(count);
    count++;
}

Output:
1
2
3
The while Statement

```java
int count = 1;
while (count <= 3) {
    System.out.println(count);
    count++;
}
```

Output:

1
2
3

Increment count
int count = 1;
while (count <= 3) {
    System.out.println (count);
    count++;
}

Output:
1
2
3
If the condition of a `while` loop is false initially, the statement is never executed.

```java
int count = 8;
while (count <= 3) {
    System.out.println (count);
    count++;
}
```

- Therefore, the body of a `while` loop will execute zero or more times.
Example: Input validation

```java
System.out.println("type in a number >5");
int num = scan.nextInt();
while (num <= 5)
{
    System.out.println ("Please try again");
    System.out.println ("type a number >5");
    num = scan.nextInt();
}
• Usually, the body of this while loop will execute zero times
```
GPA problem: Algorithm for next solution

more = true;

while (more)
{
    input qp
    input credits
    qpa = qp/credits
    print qpa

    print "Enter 1 to continue, 0 to quit "
    input answer
    more = (1 == answer)
}
int qp;
int credits;
double gpa;
Scanner scan = new Scanner(System.in);

boolean more = true;

while (more)
{
    // get input
    System.out.print("\nEnter Quality Points > ");
    qp = scan.nextInt();

    System.out.print("Enter Credits > ");
    credits = scan.nextInt();

    ... // other logic goes here
    ...
    System.out.print("Enter 1 to continue, 0 to quit > ");
    more = (1 == scan.nextInt());
}
Homework

• Read Section 5.4
  – *Always* do all self-review exercises when you review material
  – *This time it is particularly important!*

• Do Exercises EX 5.7 – 5.11