Loop - 2

Cheng, Wei  COMP110-001  May 21, 2014
Review of Previous Lecture

• Q1 What output is produced by the following statements?

```java
int counter = 0;
int sum = 0;

while(counter <= 100) {
   counter = counter + 2;
   if (counter % 3 != 0) {
      continue;
   }
   sum = sum + counter;
}
System.out.println(sum);
```
Review of Previous Lecture

• Q2 Write a program that maintains the balance of an account:
  
  – ask for a balance-update from user in each iteration
    • Positive value - deposit
    • Negative value – withdraw
  
  – If the balance-update is 0 or the balance goes below 0, exit from loop and print out the remaining balance
Today’s Contents

- For-Loop
- Nested loop
Some short-forms

\[ i = i + 1; \quad i = i - 1; \]

Can be written as:

\[ i++; \quad \text{Or} \quad ++i; \quad i--; \quad --i; \]

(we will use this frequently)

\[ i = i + j; \quad i = i + 10; \]

Can be written as:

\[ i += j; \quad i += 10; \]
Elements of loop - Again

• Loop - part of program that repeats

• Body - statements being repeated

• Iteration - each repetition of body

• Stopping condition
while loop

while (Boolean_Expression) {
    Loop body statements...
}

Evaluate Boolean Expression

- true: Execute Body
- false: End loop
for loop

\[
\text{for (Initialization; Execution\_Condition; Update) \{ }
\]

\[
\quad \text{Body\_Statement}
\]

\[
\text{\}}
\]

Evaluate

Boolean Expression

Execute

Initializing Action

Execute Body

true

false

Execute Update Action

End loop
Counting Down

```java
for (countDown = 3; countDown >= 0; countDown--)
{
    System.out.println(countDown);
    System.out.println("and counting.");
}
```
Sum of 1..n

```java
int sum = 0;

for( int i = 1; i<=n; i++) {
    sum += i;
}

System.out.println("The sum is "+sum);
```
double factorial = 1;

for( int i = 1; i <= n; i ++ ) {
    factorial *= i;
}

Sum of even numbers

```c
int sum = 0;

for( int i = 2; i<=n; i+=2 ) {
    sum += i;
}
```
For-loop

• The most common usage of for-loop is with a numerical control variable that changes constantly in each iteration:
  – E.g.: for(int i = 1; i<100; i++)

• For-loop is actually a very general form. You can express many different types of logic with it.
Simple input checking

Scanner s = new Scanner(System.in);
int input = -1;

while (input < 0 || input > 100) {
    System.out.println(“Please input a score between 0 and 100”);
    input = s.nextInt();
}

The few lines will keep reading user input until we have a number between 0 and 100.
Simple input checking – with for-loop

```java
Scanner s = new Scanner(System.in);
int input = -1;

for ( ; (input < 0 || input > 100); input = s.nextInt() ) {
    System.out.println("Please input a score between 0 and 100");
}
```

The few lines will keep reading user input until we have a number between 0 and 100.
Review – How do we end a loop?

• If you know number of loop iterations?
  – Count-controlled loops
  – for(count = 0; count < iterations; count++)

• Update variable to change the value of execution condition
  – while (input < 0 || input > 100 )
  – for ( ; (input < 0 || input > 100); input = s.nextInt() )

• Jump out of loop with “break;” statement
  if ( Integer.MAX_VALUE / factorial < counter ) {
    System.out.println(“We have to stop before it explodes!”);
    break;
  }

What does the following statements do?

```java
for (int i = 0; i < 10; i++) {
    for (int j = 0; j < 10; j++) {
        System.out.print("*");
    }
    System.out.println();
}
```
Nested Loop

What does the following statements do?

```java
for (int i = 0; i<10; i++) {
    for (int j = 0; j<=i; j++) {
        System.out.print("*");
    }
    System.out.println();
}
```
Let’s print out a Multiplication Table

```java
for (int i = 1; i < 10; i++) {
    for (int j = 1; j <= i; j++) {
        System.out.print(i + "*" + j + "=" + (i * j) + "\t");
    }
    System.out.println();
}
```

- Optional: Eclipse can show us the intermediate variable value
Brute-force coin change

Give all change solutions using 25c, 5c, 1c coins

```java
int solutionCount = 0;
int total = s.nextInt(); // total change in cents
for(int q25 = 0; q25 <= total/25; q25++) {
    for(int q5 = 0; q5 <= total/5; q5++) {
        for(int q1 = 0; q1 <= total/1; q1++) {
            if ( q25 * 25 + q5 * 5 + q1 * 1 == total ) {
                solutionCount ++;
                System.out.println( "Solution " + solutionCount + ":" + q25 + " 25c, " + q5 + " 5c, " + q1 + " 1c ");
            }
        }
    }
}
```
Multiple Inner loops

```java
for (int i = 0; i<10; i++) {
    for (int j = 0; j< 10 - i; j++) {
        System.out.print(" ");
    }
    for (int j = 0; j<=i; j++) {
        System.out.print("*");
    }
    System.out.println();
}
```

What is the output?

Can you rewrite this using one inner loop?
Off-by-one errors

for (count = 1; count < 10; count++)
– Loop 9 times

for (count = 1; count <= 10; count++)

for (count = 0; count <= 10; count++)

for (count = 0; count < 10; count++)
Finding errors

• Error checking
  – System.out.print(variable);
  – Run on simple input

• Debugger (for CS students only)
  – Eclipse: breakpoint + variable watch