Flow of Control – Branching 2

Cheng, Wei  COMP110-001  May 19, 2014
Review of Previous Lecture

• If ... else ...
  – Q1: Write a small program that
    • Reads an integer from user
    • Prints “Even” if the integer is even
    • Otherwise, prints “Odd”

• Boolean Expression & Comparison
  – Q2
    • How to compare values of primitive types?
    • How to compare objects?
  – Q3: Write out the boolean expression for testing a leap year
    • The year is evenly divisible by 4;
    • If the year can be evenly divided by 100, it is NOT a leap year, unless;
    • The year is also evenly divisible by 400. Then it is a leap year.
Gotcha ==

- `var1 = var2` (assignment statement)
  - Error!!!!!!!
- `var1 == var2` (boolean expression)

- Do NOT use `==` to compare Strings
  - `string1 == string2` //BAD
  - `string1.equals(string2);` //GOOD
Gotcha (Syntax)

if (boolean expression); // DO NOT DO THIS!!!!!!!

if (boolean expression) {
  ....
}
else {  // NO MORE boolean expression here
  ....
}

Today’s Contents

• Nested if...else...
More choices

- So far we have been working on problems with two choices (two-choose-one)

- What if we have more choices?

- Write a program that takes as input your year in college (as an integer) and outputs your year as freshman, sophomore, junior, senior, or super senior
Flow-Chart

Prompt user for year

Which year?

1. freshman
2. sophomore
3. junior
4. senior

Next step
Solution

- Translate N-choose-1 to multiple 2-choose-1 operations

Prompt user for year

Freshman

Sophomore

Junior

Senior
if (year == 1) {
    System.out.println("Freshman");
} else {
    if (year == 2) {
        System.out.println("Sophomore");
    } else {
        if (year == 3) {
            System.out.println("Junior");
        } else {
            System.out.println("Senior");
        }
    }
}
if (year==1) {
    System.out.println("Freshman");
} else {
    if (year==2) {
        System.out.println("Sophomore");
    } else {
        if (year==3) {
            System.out.println("Junior");
        } else {
            System.out.println("Senior");
        }
    }
}
Write the nested if..else.. In Java

if (year==1) {
    System.out.println("Freshman");
} else {
    if (year==2) {
        System.out.println("Sophomore");
    } else {
        if (year==3) {
            System.out.println("Junior");
        } else {
            System.out.println("Senior");
        }
    }
}
Parallel Tests

- These are four strictly parallel & mutually exclusive cases.
- We can simplify our code a bit:

```java
if (year == 1) {
    System.out.println("Freshman");
} else if (year == 2) {
    System.out.println("Sophomore");
} else if (year == 3) {
    System.out.println("Junior");
} else {
    System.out.println("Senior");
}
```
Parallel Tests

In this form, we can easily expand the choices.

```java
if (year==1) {
    System.out.println("Freshman");
} else if (year==2) {
    System.out.println("Sophomore");
} else if (year==3) {
    System.out.println("Junior");
} else if (year==4) {
    System.out.println("Senior");
} else {
    System.out.println("Super senior");
}
```
Parallel Tests

• Order of tests is important when they are not mutually exclusive
• What’s the problem with the code below?

```java
int x;
...
if (x % 2 == 0) {
    System.out.println("Multiple of 2");
} else if (x % 3 == 0) {
    System.out.println("Multiple of 3");
} else if (x % 4 == 0) {
    System.out.println("Multiple of 4");
}
```
Parallel Tests

- Order of tests is important when they are not mutually exclusive
- Order fixed:

```java
int x;
...
if (x % 4 == 0) {
    System.out.println("Multiple of 4");
} else if (x % 3 == 0) {
    System.out.println("Multiple of 3");
} else if (x % 2 == 0) {
    System.out.println("Multiple of 2");
}
```
Parallel & Mutually Exclusive Choices

- If the choices are mutually exclusive, we can write them out as a list of if-only statements.
- What’s bad about this?

```java
if (year==1) {
    System.out.println("Freshman");
}
if (year==2) {
    System.out.println("Sophomore");
}
if (year==3) {
    System.out.println("Junior");
}
```
Non-parallel Cases

- The general form. Where does the non-parallelism come from?

```
Statements

Outer Test

Statements

Inner Test

Statements

Statements
```
Non-parallel Cases

if (year==1) {
    System.out.println("Freshman");
} else {
    System.out.println("Great! You have passed your first year");
    if (year==2) {
        System.out.println("Sophomore");
    } else {
        System.out.println("Great! You have passed your second year");
        if (year==3) {
            System.out.println("Junior");
        } else {
            System.out.println("Great! You have passed your third year");
            System.out.println("Senior");
        }
    }
}

// You cannot re-write this in the if-else-if form
Another Example

Determine the last day of a month in a year

```java
if (month == 4 || month == 6 || month == 9 || month == 11) {
    maxDay = 30;
} else if (month == 2) {
    boolean isLeapYear = (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
    if (isLeapYear) {
        maxDay = 29;
    } else {
        maxDay = 28;
    }
} else {
    maxDay = 31;
}
```
One More Example

```c
int a, b, c, max;
...
if (a > b) {
    if (a > c) {
        max = a;
    } else {
        max = c;
    }
} else {
    if (b>c) {
        max = b;
    } else {
        max = c;
    }
}
```

Find the max of three
Other tips

• Indentation in code makes it easy to read
  – Corresponds to “level” in code logic
• Eclipse can automatically fix indentation:
  – Try Source -> Correct Indentation & Format
The Conditional Operator (Optional)

• Short-cut for if-else statements with return value
  \((\text{boolean}\_\text{expression} \ ? \ \text{value1} : \ \text{value2})\)

• If true, return value1. Otherwise return value2

• Example 1:
  
  ```java
  if (n1 > n2){
      max = n1;
  } else {
      max = n2;
  }
  ```
  
  can be written as
  
  ```java
  max = (n1 > n2) ? n1 : n2;
  ```
The Conditional Operator (Optional)

• Short-cut for if-else statements with return value
  \[(\text{boolean_expression} \ ? \ \text{value1} \ : \ \text{value2})\]

• If true, return value1. Otherwise return value2

• Example 2:
  
  ```java
  if (a > b){
      System.out.println("a is bigger");
  } else {
      System.out.println("b is bigger");
  }
  ```

  can be written as

  ```java
  System.out.println( ( a>b ? "a is bigger" : "b is bigger" ) );
  ```
Next Class

• Lab 2,3
• Bring your laptop & textbook