Conditional Structures and Relational Operators

Deborah A. Trytten

CS 1323/4

Jargon Reminder

- When Turingscraft says “String name has been declared,” they mean

```
String name;
```

- When Turingscraft says “int size has been declared and given a value,” they mean

```
int size = 7;  // or some other number that you
               // can’t see
```

- When Turingscraft says “Scanner stdin has been declared and constructed,” it means

```
Scanner stdin = new Scanner(System.in);
```

Relational Operators

- < Less than
- > Greater than
- <= Less than or equal to
- >= Greater than or equal to
- == Equality (pronounced equals equals)
- != Not equals
- Examples:

  3 < 7

  4 != 6

  5 > 5

  4 <= 4

iClicker Question

- Which of the following statements will be true if `size` stores the value 5?

Answer a: `size < 10`

Answer b: `size > 10`

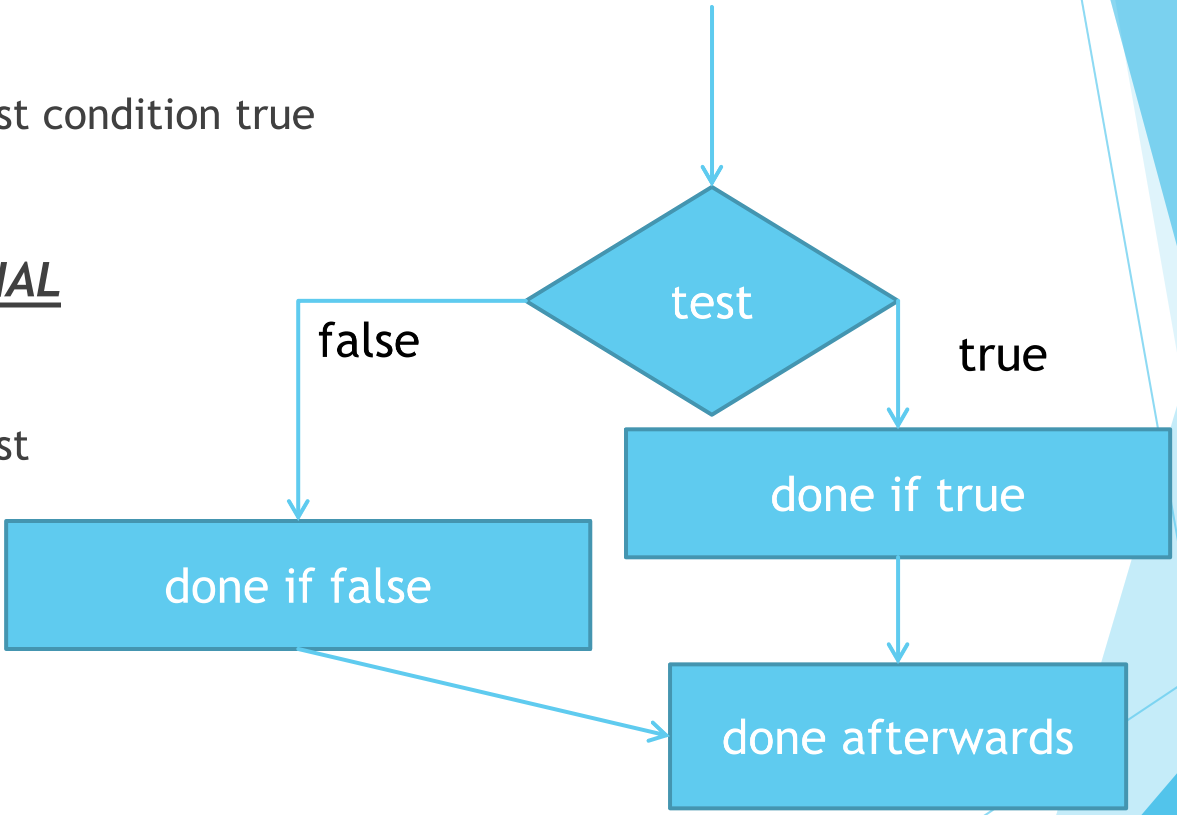
Answer c: `size >= 10`

Answer d: `size == 10`

Answer e: `size != 5`

Conditional Statements

```
if (test condition)
{
    // Done if test condition true
}
else // OPTIONAL
{
    // Done if test
    // false
}
```



Write Code: Conditional Example

Write a code fragment that uses a conditional statement to print either “the price of tickets is going up” or “the price of tickets is not going up” depending on how today’s price compares with yesterday’s price.

Assume the prices are stored in the following variables:

```
double todayPrice;
double yesterdayPrice;
```

Think, Pair, Share

- Print out a warning if a non-positive value is entered
  - Examples:
    - You may not order -1 items
    - You may not order 0 items
  - Note: Non-positive is different than negative

Announcements and Reminders (Sept 9)

- Upcoming assignments:
  - Project 2 due tonight
  - zyBook Ch.6 due tomorrow

- More Turing's Craft difficulties
  - TC 4 has at least one broken problem
  - I will remove it later today

- Career fair
  - Wednesday and Thursday this week
  - Thursday is for engineers

Review and Preview

- Last time:
  - Relational operators
    - <, <, <=, >=
    - Result is a boolean value (3 < 5 is true)
  - Equality operators
    - ==, !=
    - Result is also a boolean value (3 == 5 is false)
  - Conditional statements
    - if statement
    - optional else clause
- Today:
  - Block statements (using curly braces with conditionals)
  - Precedence of relational and equality operators
  - Comparing Strings for equality
  - Cascading conditionals

Block Statements

- Java uses statement boundaries to determine which statements are executed as part of a conditional statement
  - No curly braces—one statement
  - Curly braces can be used to group statements together into a block
  - Extra curly braces don’t hurt anything

- Indentation does not matter to Java
  - But it matters deeply to me, the TAs and anyone and everyone who reads your code

Example

- Suppose we want to set shipping, tax, and total to zero if `numberOrdered` is negative

```
if (numberOrdered < 0)
     tax = 0;
     shipping = 0;
     total = 0;
```

- What is wrong with that code?

Precedence

- Suppose we want to make sure that the value m is odd
- Concept: Add one to even numbers

```
if (m%2 == 0) {
   m = m + 1;
}
```

- Do we need () around `m%2`? Which gets applied first, % or == ?

Precedence

- General goal of Java precedence is to eliminate the need for parentheses
  - Does not necessarily make it good code

- Which of the lines of code below could make sense?

```
a<b  ==  c<d

a==b  <  c==d
```

- Rule: relational operators have higher precedence than equality operators

Precedence (so far)

1. Parentheses
2. Unary: `+, -`
3. Casts: `(int) (double)`
4. Multiplicative: `* / %`
5. Additive (Binary): `+ -`
6. <u>Relational</u>: `< > <= >=`
7. <u>Equality</u>: `== !=`
8. Assignment: `=`

iClicker Question

- What value does the statement below have when time’s value is 3.85
  - Consider precedence
- `time + 3 <= 9`

- Answer a: true
- Answer b: false
- Answer c: It is an error
- Answer d: 6.85

Think Pair Share

- Some companies give customers special bonuses for joining their loyalty program
  - goldStatus // Choose type

```
int bonusPoint;

double saleDollars; // Given value elsewhere
```

- Write statements that give regular customers one bonus point for every $10 purchased, and gives gold customers one bonus point for every $5 purchased
  - Hint: Work with some numeric examples to figure out the operations

Test Strings for Equality

```
String pointer = new String("Jazz");
String labMix = new String("Raven");
String oldDog = new String("Raven");

System.out.println(pointer.equals(oldDog));
System.out.println(oldDog.equals(labMix));
System.out.println(pointer == oldDog);
System.out.println(oldDog == labMix);
```

Equals() and ==

- Strings are objects, not primitive data types
  - == works with primitive data types
- When we compare the contents of two Strings, we have to use .equals() or .equalsIgnoreCase()
  - Called deep comparison
  - Compares the contents of the String instead of the references
- Show memory diagram

iClicker Question

```
String first = new String("Monday");
String second = new String("Monday");
```

Which of the following is true?

a) `first.equals(second)`

b) `first == second`

c) `first != second`

d) Both a and c are true

Announcements and Reminders (Sept 11)

- Upcoming assignments:
  - Homework 2 due Thursday (tomorrow)
  - TC 4 and TC 5 due Friday
  - Midterm 1 next Monday
    - Old midterms posted on Canvas
- Additional tutoring hours posted on syllabus and Canvas
- Career fair today and tomorrow

Cascading Conditionals

- Often used for cases that are mutually exclusive and collectively exhaustive
  - When we have more than two choices, we can combine else statements with an if inside an else

```
int x; // given a value elsewhere
if (x < 0)
   System.out.println("X is negative");
else if (x==0)
   System.out.println("X is zero");
else
   System.out.println("X is positive");
```

- Notice indentation

Example

- Write a code fragment that prints a free gift depending on the value of a purchase
  - $0-$9.99: a free pen
  - $10-$99.99: a coaster
  - $100-$199.99: ear buds
  - More than $200: portable speaker

Think, Pair, Share

- Write a nested if statement that prints out whether a student is a first year, second year, third year or fourth year based on the number of college credits completed
- The rules are below
  - A student is a first year if they have less than 30 hours
  - A student is a second year if they have 30 or more, but less than sixty hours
  - A student is a third year if they have 60 or more, but less than 90 hours
  - A student with 90 or more hours is a fourth year student

iClicker Question

- If burritos has the value 9, what value is assigned to burritos in the computation below?

```
if (burritos > 6) {
    burritos = 3;
}
else if (burritos >= 3) {
    burritos = 5;
}
else {
    burritos = 8;
}
```

a) 9 b) 3 c) 5 d) 8

Dangling else

```
if (x == 3)
   System.out.println("x is 3");
if (y == 7)
   System.out.println("y is 7");
else
   System.out.println("Oh, no!");
```

- Who owns the else?
  - How do you fix that if it isn’t what you need?

Common, hard-to-spot mistake

```
int x = 1;
int y = 2;

if (x == 100);
{
    y = 3;
}
```

- What value is stored in y, 2 or 3?