Lesson 5-1: If Statements and Logical Operators

Computer Science 46A: Introduction to Programming

San José State University

Announcements

- Homework #3 will be posted
- Lab #5 is this Friday (2/28)
- Midterm 1 – Mar 27
- Midterm 2 – May 6
- Finals May 14 – 10:45am-12:45am

Learning Objectives

By the end of this lesson, you should be able to:

1. Use `if`, `else if`, and `else` statements to run different pieces of code depending on provided input
2. Use relational operators to compare different values

The `if` statement

- The if statement is one of the most powerful and commonly used statements of any programming language
- It allows you write programs that can handle a variety of data types
- Syntax:

```
if (condition)
{
    // when condition is met, this code block
    // will be run; otherwise nothing happens
}
```

A note about Syntax

- If statements *can* be written using a shorthand
- If there is only one line in your if statement, you do not need brackets
- These are the same:

With brackets:

```
if (condition)
{
   System.out.println(“One line”);
}
```

Without brackets:

```
if (condition)
   System.out.println(“One line”);
```

Style Alert: `if` statements have brackets

In this class, write your `if` statements with brackets

With brackets: 😃

```
if (condition)
{
    System.out.println(“One line”);
}
```

Without brackets: 😡

```
if (condition)
    System.out.println(“One line”);
```

Later, when you are professionals, you or your team might adopt the short style
For this class, since it is more clear, we’ll use brackets

The `else` statement

The `else` statement provides an alternative to the `if` statement

Syntax:

```
if (condition)
{
   // when condition is met, this block will run
}
else
{
   // when condition is not met, this block will run
{
```

Example: IntCompare

```
if (num1 > num2)
{
    System.out.println(num1 + " is bigger than "+num2);
}
else
{
    System.out.println(num1 + " is less than or equal to "+num2);
}
```

Poll Everywhere: Question 1

Consider the following code:

```
if (num1 > num2)
{
    System.out.println(num1 + " is bigger than "+num2);
}
else
{
    System.out.println(num1 + " is less than or equal to "+num2);
}
```

What is the input if `num1 = -5, num2 = -5`?

A) num1 is bigger than num2

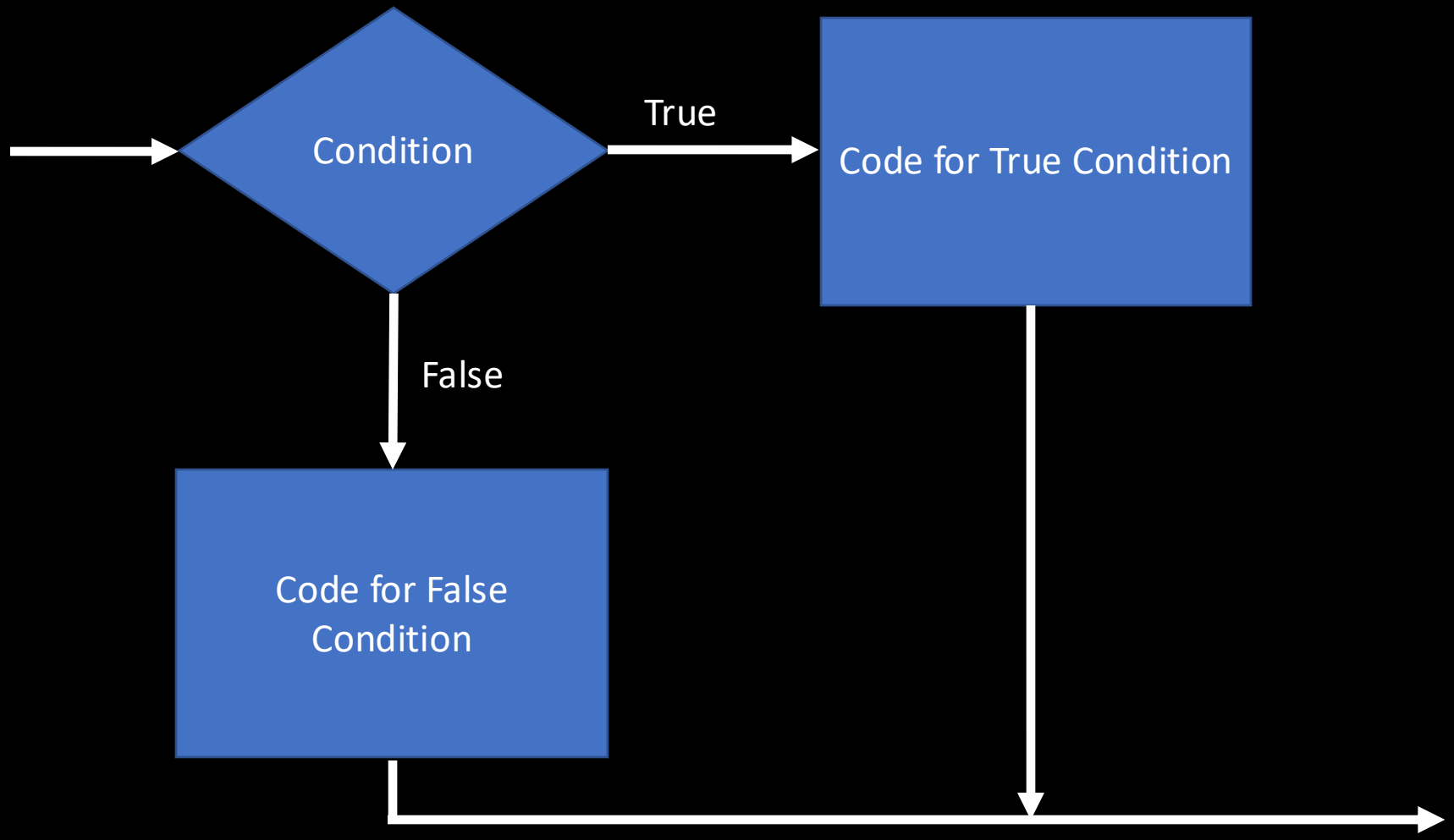
B) -5 is bigger than -5

C) num1 is less than or equal to num2

D) -5 is less than or equal to -5

PollEv.com/narayanbalasubramanian644

If-else Statement Diagram



Conditions for the `if` statement

Named after a math guy with the last name Boole

All conditions for the if statement must be `boolean`

i.e. they must either be true or false

To compare values, we use relational operators:

Operator	Description
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
==	Equal to
!=	Not equal to

Careful with numbers that have many decimals!

- From math: $(\sqrt{3})^2 = 3$
- In Java: Try this in `DoubleCompare.java`

```
double1 is 3.0
double1 = 3.0
double2 is Math.sqrt(3)*Math.sqrt(3)
double2 = 2.9999999999999996
double1 is not equal to double2
```

Each time an operation with decimals is carried out, only 15 digits are stored – we lose information at each turn!

Comparing Strings

- Two strings are equal if all of their characters are equal
- To compare strings, we use the string method `.equals()`:

```
string1.equals(string2);
```

- We CANNOT use the operator ==
  - Try this in `StringCompare.java`
- The == operator checks whether both strings are in the same memory location (for example, if they are both initialized with the same value);
  - It does not check char by char to see if they all match

The `else if` statement

- What happens if there is more than one option?
- An `else if` block:

```
if (conditionA)
{
    // when conditionA is met, this block will run
}
else if (conditionB)
{
    // when conditionB is met, this block will run
{
else
{
    // when conditionA is not met, and
    // condition B is not met, this block will run
{
```

An else-if block is added between the if and else block

We can add as many as we need!

Example: YearToHSTitle

A series of `if`, `else if`, and `else` statements allow our program to produce different outputs depending on the inputs provided

```
Enter a year in 4-year high school: 1
A first year is called a Freshman
Enter a year in 4-year high school: 2
A second year is called a Sophomore
Enter a year in 4-year high school: 3
A third year is called a Junior
Enter a year in 4-year high school: 4
A fourth year is called a Senior
Enter a year in 4-year high school: 5
Unexpected year for a 4-year school: 5
```

Example output for YearToHSTitle

Poll Everywhere: Question 2

Consider the following output:

```
Enter a year in 4-year high school: 1
A first year is called a Freshman
Enter a year in 4-year high school: 2
A second year is called a Sophomore
Enter a year in 4-year high school: 3
A third year is called a Junior
Enter a year in 4-year high school: 4
A fourth year is called a Senior
Enter a year in 4-year high school: 5
Unexpected year for a 4-year school: 5
```

The code in `YearToHSTitle` has 1 `if` statement and 1 `else` statement. How many `else if` statements are there?

A) 0

B) 1

C) 3

D) 5

PollEv.com/narayanbalasubramanian644

Participation Exercise 5-1a: FavoriteColor

Goal: Write a program that takes in a color giving one response if its blue, and a different response if its any other color (or word)

Sample output #1

```
Enter your favorite color: blue
Mine, too
```

Sample output #2

```
Enter your favorite color: green
That's pretty, too
```

Hint: Be sure to use the Scanner method to take a string as an input

Codecheck Link: HERE and on Canvas

Participation Exercise 5-1b: ConvertingGrades

Goal: Write a program that turns a letter grade into the quantity that contributes to your GPA

Sample output

```
Enter the letter grade: A
The numeric grade for letter grade 'A' is 4.0.
Enter the letter grade: B
The numeric grade for letter grade 'B' is 3.0.
Enter the letter grade: C
The numeric grade for letter grade 'C' is 2.0.
Enter the letter grade: D
The numeric grade for letter grade 'D' is 1.0.
Enter the letter grade: F
The numeric grade for letter grade 'F' is 0.0.
Enter the letter grade: E
The numeric grade for letter grade 'E' is -1.0.
```

Expected output for ConvertingGrades

Codecheck Link: HERE and on Canvas