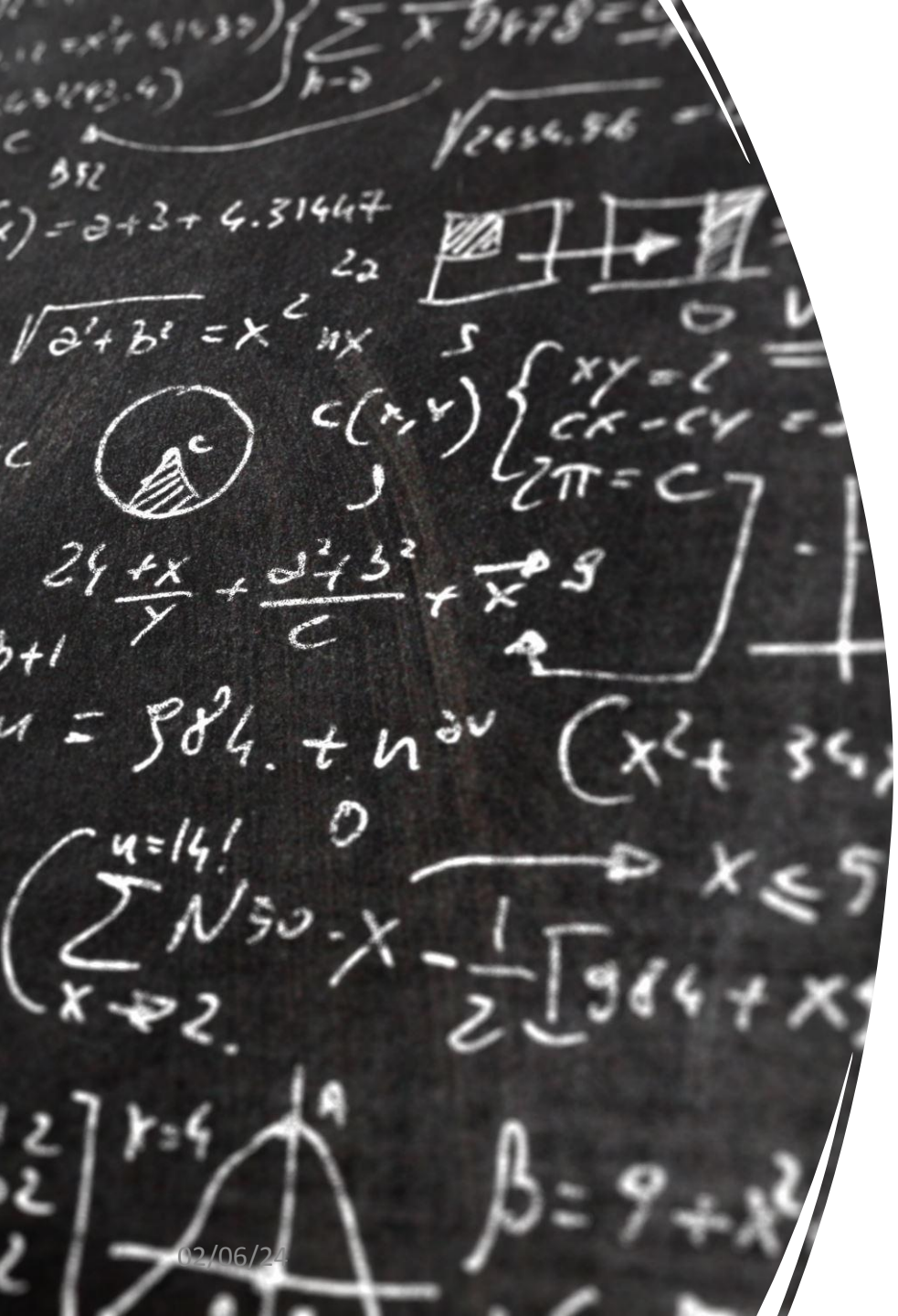


CS 113 – Computer Science I

Lecture 05 – Methods II

Tuesday 02/8/2024

Announcements

If you didn’t create a Github account, do it right now.

HW1 is released, due Thursday Feb 8th

Roster

Mentimeter: 7592 5337

Overview

- Github recitation
- Methods review
- Start boolean conditionals

Github

Collaborative environment with version control

Github

1. Create a repo
2. Follow the instructions
    a. refresh repo to see your changes
3. Create cs113 folder with a file in it
    a. `git status`
    b. add, commit, and push
4. Exit the goldengate server
5. Clone the repo (locally)
    a. Run “git log”

https://rogerdudler.github.io/git-guide/

Github

Oh no I deleted a file!

```
rm Lab0.java
git status
git checkout -- Lab0.java
git status
```

Our file is back!

Github - merge conflicts

1. Make a change locally
    a. `git status`
    b. `git add, commit, push`
2. Log into goldengate
    a. Look at the file you modified. It shouldn't include the modification.
    b. make a different change in the same location
    c. `git status`
    d. `git add, commit, push`
    e. oh no! We're out of sync
    f. `git pull` //sync the other changes
    g. `git config pull.rebase false`
    h. manually resolve the conflict
    i. `git add, commit, push`

git log :)

Methods

What’s the purpose of a method?

Anatomy of a method

- All methods have the following things:
  - Name
  - Parameter
  - Body
  - Return Type

```
public static int method1 (int param1,
                           String param2) {
  /**
    body of the method
  */
  return 0;
}
```

Scope

- area of a program where a variable can be used

- Stack diagram's helpful for identifying scope

Code

Scope

```
public class Area {

    public static double area(double width, double height) {
        float result = width * height;
        return result;
    }

    public static void main(String[] args) {

        double size = area(10.0, 5);
        System.out.printf("Area is %d\n”, size);
    }
}
```

Booleans and Conditionals

Motivation

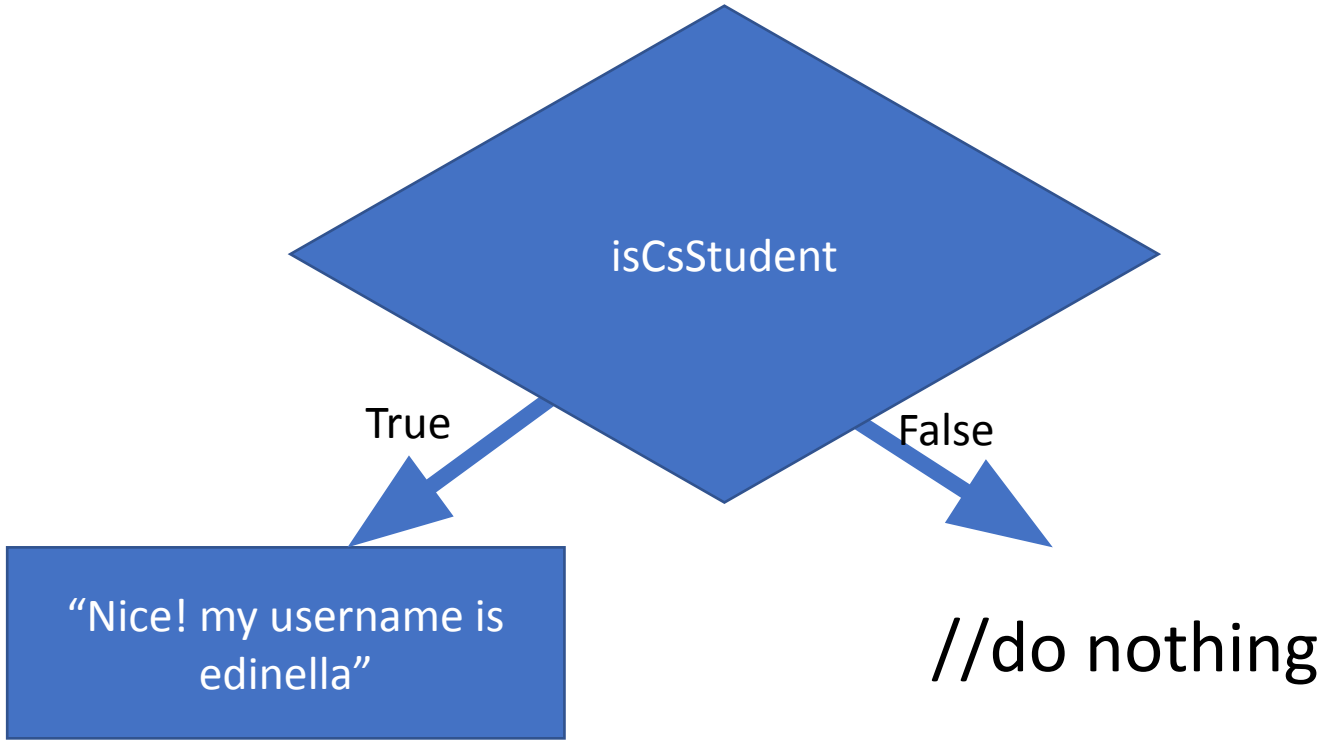
Write a program to print your username **<u>if the usr is a cs student</u>**

A new data type: Booleans

- Contains two possible values:
  - true; false;
  - bool isWet = true;
- boolean expression

Decision making

Idea: Branching decision-making based on Boolean expressions



Logical Operators

- Way to combine Boolean expressions

- logical Operators:
  - && - and
  - || - or
  - ! - not

Rules of logical operators

1. X && Y is true when
    1. Both X and Y are true

2. X || Y is true when
    1. X is true or Y is true

3. !X is true when
    1. X is false

4. !X false when
    1. X is true

Boolean Expressions & Relational Operators

- Conditional expression produces either true or false

- Relational Operators:
  - >
  - >=
  - <
  - <=
  - ==
  - !=
- Watch out about == vs =

Exercise: relational expressions

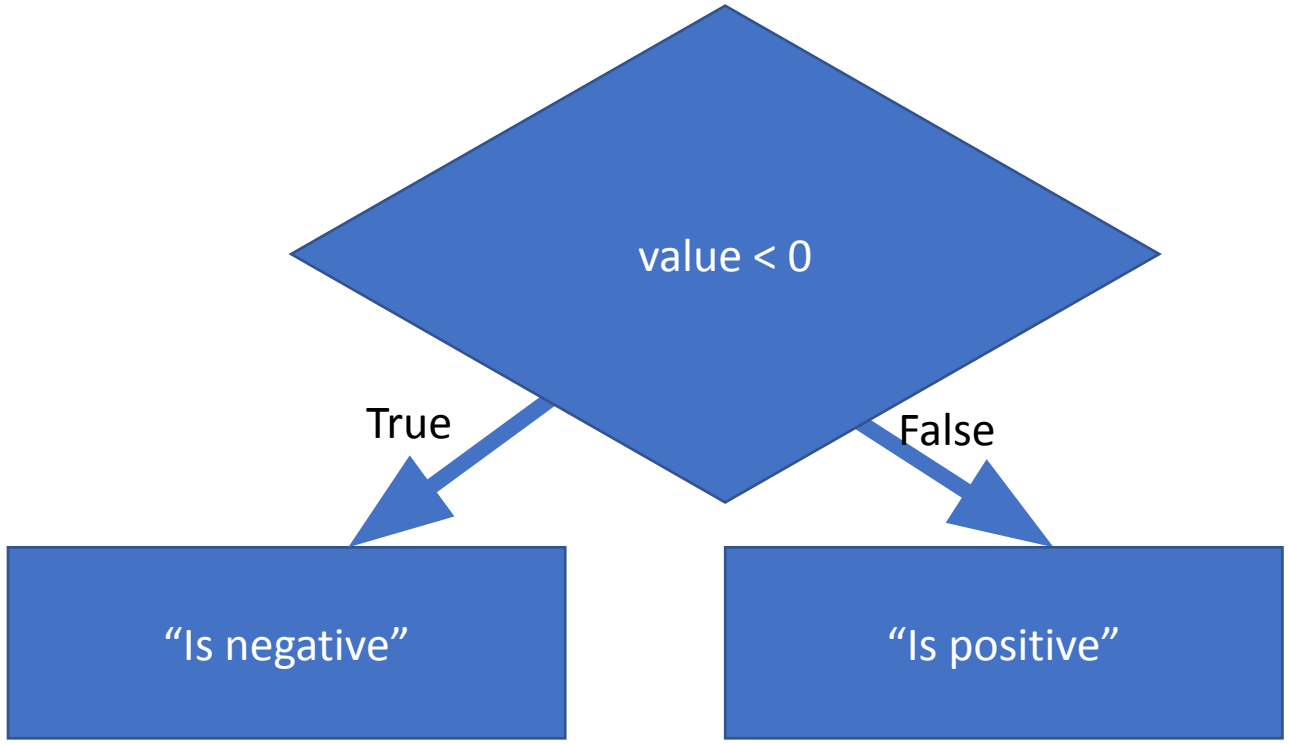
```
int temp = 68;
double val = 10.5;
boolean raining = true;
```

Expression	Value	Type
temp > 80		
val != 5.6		
val >= 10.1		
raining == true		
raining		
raining == false		

`If` syntax

```
if (condition) {
    //statements
} else {
    //statements
}
```

Decision making



Decision Making

Write a program that asks if they're happy
asks a user if they know it

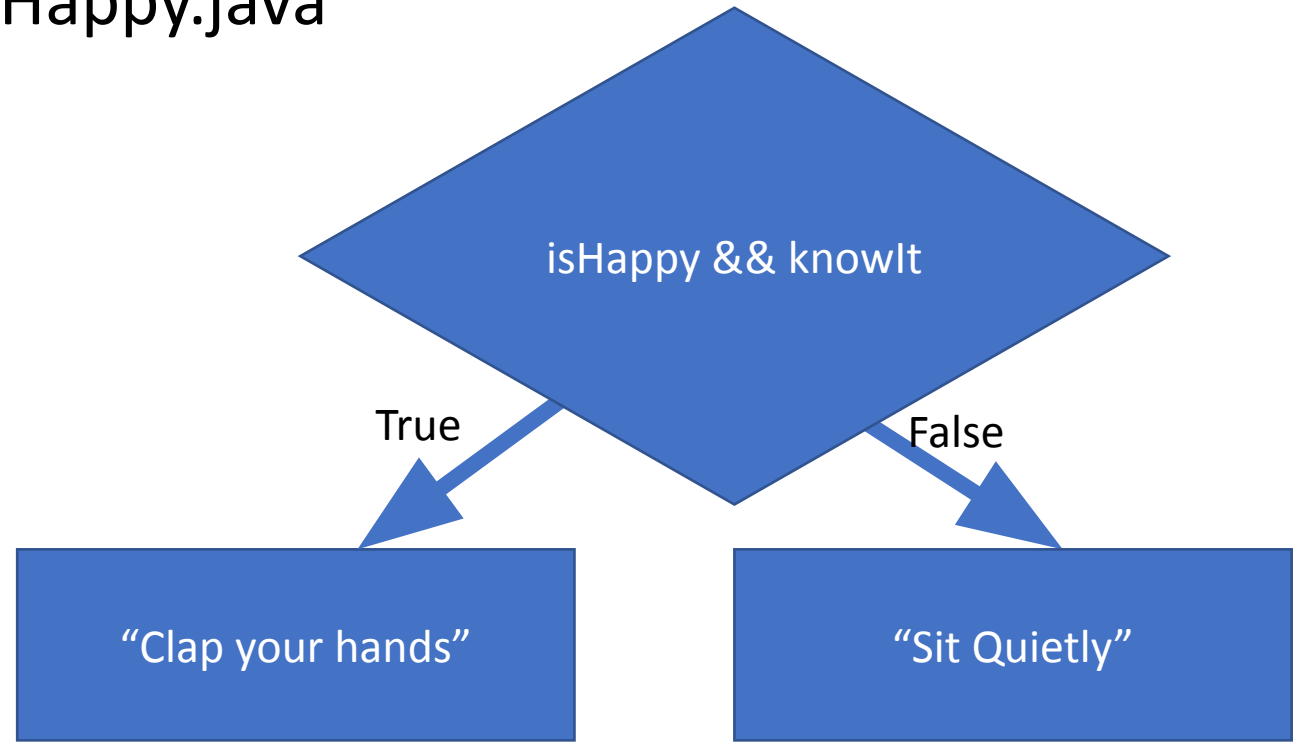
If they're happy and they know it
    -> print "clap your hands"

else
    -> print "sit quietly"

Decision making

Idea: Branching decision-making based on Boolean expressions

- Example: A **decision tree** for Happy.java



Exercise: logical expressions

```
boolean isHappy = true;
boolean knowIt = false;
int temp = 40;
```

Expression	Value	Type
isHappy && knowIt		
isHappy		
isHappy || temp > 80		
isHappy || knowIt		
!knowIt		
isHappy && (temp < 80 || !knowIt)		

Summary

1. Github
2. Methods
3. Boolean and conditionals