Announcements

- Nice job with Lab 3! Due today/tomorrow. Don’t forget to commit/push. Can “check” your submission on https://evolene.cs.williams.edu
- HW 3 has been posted (due Mon)
- If you have questions about Lab/HW grades, please come talk to us!
- Jeannie office hours: Wed 1-3:30, Thur 1-2:30
- Looking ahead: Lab 4 is challenging! Please read lab in advance and develop implementation plan

Last Time

- Binary communication
- Manchester encoding vs on/off keying
- …we’ll return to that again soon!

Today’s Plan

- Fun with Strings!
- While loops
String methods()

- For String str:
  - str.length() : int \( \geq 0 \)
  - str.startsWith(otherString) : boolean
  - str.equals(otherString) : boolean
  - str.indexOf(otherString) : int \( \geq -1 \)
  - str.indexOf(otherString, startIndex) : int \( \geq -1 \)
  - str.contains(otherString) : boolean
  - str.substring(startIndex, endIndex) : String
  - str.substring(startIndex) : String

https://docs.oracle.com/javase/9/docs/api/index.html?java/lang/String.html

String indexing

- Indices start at 0!

amanaplanacanalpanama
012345678901234567890
111111111112

A Rose by Any Other Name
The DICT Protocol

- www.dict.org at port 2628
- Word Searching & the DICT Protocol
- See ChoiceReplacer, DictWordClient
Moving on…

- Initially all execution was linear
- When if-else statements, we introduced branching
- While loops allow us to repeat statements and "visit boxes twice" (in our flow chart)

Courtesy xkcd.com
While Loops

```java
int count = 1;
while(count <= 50) {
    System.out.println(count + " Goodbye!");
    count = count + 1;
}
```

(Run FiftyWays in BlueJ and in the terminal)

Now what happens?

Your first infinite loop!

```java
int count = 1; // initialize loop variable
while(count <= 50) {    // begin loop: check condition
    System.out.println(count + " Goodbye!");
    count = count + 1;    // Execute statements in loop body
}
```
While Loops

While Loops

```java
while ( condition ) {
    statement(s) to be repeated
}
```

While Loops

```java
statement(s) to initialize loop variable(s)
while ( condition ) {
    statement(s) to perform one step
    statement(s) to move onto next step
}
```

```java
line = toServer.in.nextLine();
while ( ! line.equals("") ) {
    message.append( line + "\n" );
    line = toServer.in.nextLine();
}
```