CompSci 4
Java (Part 2)
Nov 20, 2008

Prof. Susan Rodger
Announcements/Review

• Assignment 7
  – Storyboard due Nov 25, Alice world Dec 2

• What is Eclipse?
  – Environment to aid you in writing Java code
  – Create ONLY one project for each classwork, think of it as one Alice world with several classes in it
  – Put each new class in that same project

• What is an APT?
  – Web-based tester for one method at a time

• Today – More APTs
  – String parts, arrays
Review Strings

- String word = "CompSci 4";
- `word.length()` – returns length of string
- `word.toCharArray()` – returns string as an array of characters
- `word.charAt(5)` – returns character at position 5
- Loop over characters in a string
  ```java
  for (char ch: word.toCharArray())
  {
      // do something to ch
  }
  ```
More on Strings

• `word.indexOf(“Sci”)`
  – Returns first position of “Sci” in “word” or –1 if not in word

• `word.substring(4,6)`
  – Returns part of string in “word” that starts at position 4, goes up to but not including position 6 and is of length 2

• `word.substring(4)`
  – Returns part of string in “word” that starts at position 4 til the end of the string

• `word = word + “ rocks”;
  – Build a string – append to the right end
Print out a value

• System.out.println(\textit{string value});

• Prints out on one line

• Example:
  
  \begin{verbatim}
  System.out.println("word is ");
  System.out.println(word);
  \end{verbatim}

What is printed?
Example

```java
String course = "CompSci 4 Spring 2006";

System.out.println(course);

int pos = course.indexOf("Spring");
String part1 = course.substring(0, pos);
String part2 = course.substring(pos+6);
course = part1 + "Fall" + part2;

System.out.println(course);
```

How could you change 2006 to 2008 in course?
if – else if - else

- Alternative to nesting ifs
- Can have as many “else if” as you want
- else is optional
- First case that is true is executed

- See example on next page, what happens when num=3? num=6? num=10?
What is output for values of num?

```java
if (num > 8)
{
    System.out.println(num);
}
else if (num > 5)
{
    System.out.println(num);
}
else
{
    System.out.println(num);
}
```
Arrays

- **Parameter:** double [] numbers
  - Means an array of doubles, name of array is numbers
- **Loop over items in an array – collections loop**

  ```java
  for (double item: numbers)
  {
    // do something with item here
    // item is one_item_at_a_time
  }
  ```
To refer to one item in Array

- Name of array[position number]
- Be careful, position number must exist!

- Example:
  numbers is an array of doubles

System.out.println(numbers[3]);
Array Example

- Find max number, assume at least one number in the array

```java
public class MaxInArrayTest {
    public double MaxInArray (double[] numbers) {
        double max = numbers[0];
        for (double num : numbers) {
            if (num > max) {
                max = num;
            }
        }
        return max;
    }
}
```
Problem: DNA Max

- Given an array of DNA strands and a nucleotide (a, c, g, or t)
- Return the strand with the most occurrences of the nucleotide
- If there is more than one strand with the max number, return the longest such strand
Example

- Given array
  [“agt”, “aagt”, “taattt”, ccatg”]
- Given nucleotide “a”
- Returns “taattt”
- “a” appears 2 times max in a strand
- Longest such strand is “taattt”
Solve this problem in Eclipse

```java
public class DNAMaxNucleotide {
    public String max(String[] strands, String n) {
        // fill in code here
    }
}
```

- What steps do we do?
- What do we already know how to do?
Classwork today – More APTs

• Test java methods using APT
• Create one new Java project called CPS4Sec1Nov20 (or Sec2) for all classwork
• Create three new classes based on APT problems – test with APT
  – Class and methods must be spelled exactly as shown
• Get checked off
Classwork Problems
see sheets on APT page for more detail

- Class: DNAComplement
  - Name of method: complement
  - Build and return a new string with complements

- Class: DNAReverse
  - Name of method: reverse
  - Build and return a new string that is the reverse

- Class: LongStrand
  - Name class: longest
  - Return string with most nucleotides