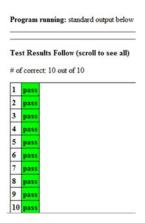
# CompSci 100e Program Design and Analysis II



January 18, 2011

Prof. Rodger

CompSci 100e, Spring2011

#### Visualizing Text

- Text Cloud aka Tag Cloud?
  - Number of occurrences/emphasis indicated by size of word
  - Great visual/statistic: http://chir.ag/phernalia/preztags/



- http://www.nytimes.com/gst/mostsearched.html?period=30&format=tagcloud
  - · What information is stored in the URL of the NYTimes site above?

#### **Announcements**

- Lab 0 was to get Eclipse/Ambient running
- Lab 1 (Jan 21/24) APTs
- APT Assignment out (do 7 APTs)
  - 2 done in class, 2 in lab, 3 on your own
  - Submit all 7 together one Java Project by Jan 25
- Consulting hours starting soon.....

CompSci 100e, Spring2011

# Lab 0: Text Clouds

- Point To install Eclipse, Ambient, Java
- Text clouds: A simple yet powerful idea
  - Visualization of most frequently occurring words within some body of text
  - Color or font size indicates word frequency



- What is involved with generating text clouds?
  - Steps? Issues?
  - See SimpleWordCount.java and SimpleCloudMaker.java

# **Problem Solving and Programming**

- How many words are in a file?
  - What's a word?
  - What's a file?
  - How do we solve this: simply, quickly, ...?
    - What's the best we can do? Constraints?
- How many different/unique words are in a file?
  - How is this related to previous task?
- How many words do two files have in common?
  - Spell-checking, stemming, Did you mean ..?
- How many codons does DNA have in common?

#### Array

Declare and initialize an array of integers

```
int[] values = new int[12];
```

• Set it to these values:

```
8 3 4 3 8 2 4 4 6 2 8 4
```

Access item in slot 6 in the array

```
values[6]
```

• Array is fixed size. The size is:

```
values.length
```

#### Java - for loop

```
public void printFencePostfor(int numberPosts) {
   String rail = "===";
   String post = "I";

   System.out.print(post);
   for (int k = 1; k < numberPosts; k++) {
        System.out.print(rail);
        System.out.print(post);
   }
   System.out.println(" ");
}</pre>
```

#### Example

```
for (int k=0; k<values.length; k++)
{
   values[k] = values[k] + values[k-1];
}</pre>
```

- What does this do?
- Is it correct?

#### Classwork

SimpleWordCount.java

CompSci 100e, Spring2011

#### Using Scanner class to read

- Import java.util.Scanner;
- Declare Scanner and bind it to a file (last slide)
- Make sure there is input still to read
   while (in.hasNext())
- Read next line

```
String line = in.nextLine();
```

Read next word/token

```
String word = in.next();
```

• Read next integer

```
String word = in.nextInt();
```

## Reading from Files

- import java.io.File;
- Declare a file

```
File fileOfCats = new
File("cats.txt");
```

• Use file – pass it as an argument to a Scanner

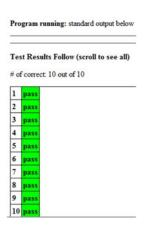
```
Scanner in = new
Scanner(fileOfCats);
```

# What will we use Eclipse for in CompSci 100e?

- Use to write complete java programs
  - Access libraries
- Use as an editor to write text files
  - README gives info about the program
- Use to write simple methods, then test with APT

## APT – Algorithmic Program Testing

- Not a complete java program
  - No main method
- Focus on and solve one small problem
- Rich set of data for testing
- Use Eclipse editor for APT, but cannot run program! Why?
- Goal: all green



#### Solve APT

DNAreverse

CompSci 100e, Spring2011

## **Strings**

- String
  - –a sequence of characters
  - -objects of the String class
- String constants:

```
"Hello, World!"
```

- All Strings are constants don't use "new" w/ String
- String variables:

```
String message = "Hello, World!";
```

• String length:

```
int n = message.length();
```

• Empty string: ""

## String

Concatenating Strings

```
-Use the + operator:
```

```
String name = "Dave";
String message = "Hello, " +
name;
```

Automatic type conversion

```
String a = "Agent00"; int n = 7;
String bond = a + n;
// bond is "Agent007"
```

# What can you do with strings?

- Look at API
- int length()
  - Returns length of string
- String substring(int beginIndex)
  - Returns substring from beginIndex to end of string
- String substring(int beginIndex, int endIndex)
  - Returns substring from beginIndex to endIndex -1

## Example

```
String one = "ferriswheel";
String two = one.substring(5);
String three =
  one.substring(4,6);
```

What are two and three?

# Finding substrings in strings

- int indexOf(String str)
  - Returns first position of str in the string
  - First position in a string is 0
- int indexOf(String str, int fromIndex)
  - Returns first position of str starting at fromIndex

# Example

```
String one = "Cotton Candy";
String two =
  one.substring(indexOf("Can"),
    indexOf("Can")+4);
```

What is two?

# Strings

- String word = "CompSci 100e";
- 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

```
for (char ch:
word.toCharArray())
  {
   }
```

#### Solve APT

ClassScores

CompSci 100e, Spring2011