

# CSSE 220 Day 25

Exam Retrospective

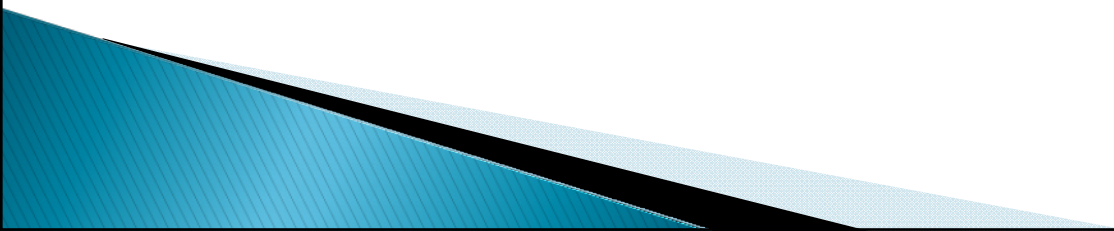
Sorting intro

Work on Spellchecker Project

# CSSE 220 Day 25



# Today's Agenda

- ▶ Exam retrospective
  - ▶ Sorting intro
  - ▶ Meet your Spellcheck partners and get organized; start your design
- 

# Exam results

## Programming Part

Score	Number of students
62+	7
54-59	10
48-53	8
42-47	3
36-41	4
30-35	3
<30	3

## Written Part

Score	Number of students
36-40	1
32-35	3
28-31	8
24-27	9
20-23	7
16-19	8
<16	3

## Total

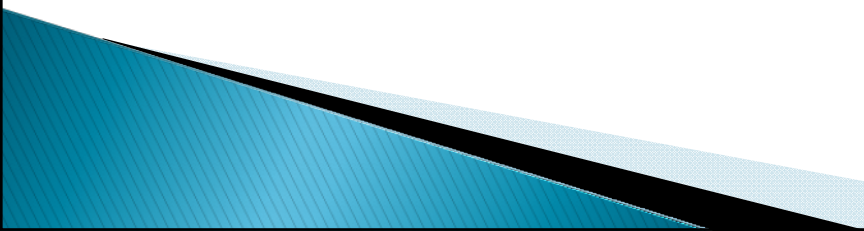
Score	Number of students
90-100	5
80-89	10
70-79	13
60-69	2
50-59	5
40-49	2
<40	2

The average on the written part of the test was a little bit low, so I lowered the percentage required for each grade in the gradebook's grading scale.

# Course grades at this point

Grade	Number of students
A	11
B+	8
B	10
C+	4
C	4
D+	0
D	4
F	0

# Written exam problems

- ▶ I will go through each problem quickly.
  - ▶ If you need more explanation of any of them, please ask questions.
- 

# Sorting Intro

- ▶ What do we mean by "sort"?
- ▶ What is the best sorting algorithm?
- ▶ The three very simple Algorithms
  - Bubble Sort
    - Why is it so slow?
  - Insertion sort
  - Selection sort
- ▶ Inversions and movement
- ▶ Faster algorithms

# SpellChecker and Suggester

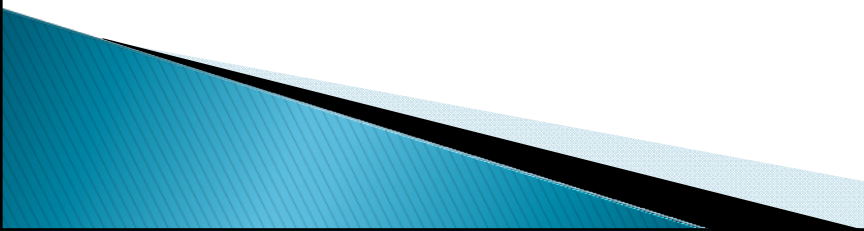
- ▶ Other projects have been highly-specified. For this one, you have a lot of leeway and can be very creative.
- ▶ GUI-based program
- ▶ Check the words of a text file for spelling
  - User can browse to file
- ▶ Flag words that are not in program's dictionary
- ▶ Suggest possible alternate spellings
  - Think of ways misspelling can occur:
    - missing or added letters
    - transposed letters
    - no space between words
    - things you come up with
- ▶ An interface that allows user to correct the spelling.
  - change, ignore, ignore all, ...



# SpellChecker and Suggester

- ▶ Some GUI things you'll want to learn how to do
  - Browse to a file and open it
  - Deal with text in a text box
  - Display a list of choices and get user selection
- ▶ Some things to do if you didn't do them already.
  - Look for a dictionary to use (share it!)
  - Look at user interfaces of some spell-checkers
  - Look up various Java classes that may be useful
    - Especially helpful: The Java Swing book from Safari Tech Books online (see course syllabus)

# Mini-project timetable

- ▶ Now. Look for a dictionary, think about the kinds of spelling errors you want to detect/correct.
  - ▶ Day 25. Begin working with your partners.
  - ▶ Day 27. Demonstrate some progress in class.
  - ▶ Day 30. Final submission of the project is due.
- 

# Spell-checker teams

220-2008-spell01	allencw	bacaep	wenzelpi
220-2008-spell02	andersar	pientsars	pridalmj
220-2008-spell03	berrygl	baekj	salisbjm
220-2008-spell04	burnsjl	bergencb	watersbt
220-2008-spell05	brumbams	reillywj	blankeaz
220-2008-spell06	dunnmp	vanderkl	junkersa
220-2008-spell07	fehribrm	sullivsd	williakl
220-2008-spell08	bottjd	buetowbp	tamal
220-2008-spell09	bussinjr	jennemj	iversopn
220-2008-spell10	goodrijk	hilljd	skaggskd
220-2008-spell11	hansenrl	kriesbsd	pickdp
220-2008-spell12	gerthwd	merseljp	kotsybja
220-2008-spell13	nowickpj	snivelee	yimah
220-2008-spell14	watersdc	thiememd	wisejl

Repositories will be available soon