Program Design and Methodology II (aka Data Structures)

theoretical and practical study of data structures, algorithms, and program design

- In theory there is no difference between theory and practice, but not in practice

Object-oriented programming and design with C++
Use of off-the-shelf libraries and home-brewed libraries: when to use, re-use, and invent

Language features and idioms; design patterns
Extrapolating beyond C++ (e.g., Java and beyond?)
Course Administrivia

- Be sure to check website and news regularly
  - http://www.cs.duke.edu/~ola/courses/cps100.html
  - duke.cs.cps100

- Grading (see web), 500 total points distributed
  - ~6 major programming assignments, 28%
  - ~5 minor programming assignments, 10%
  - written assignments, random quizzes, 16%
  - 2 tests, 26%, final exam, 20%

- In general, programming assignments are done individually, sometimes in groups of two or three

- Keep up with work, persevere.
What is Computer Science?

What is it that distinguishes it from the separate subjects with which it is related? What is the linking thread which gathers these disparate branches into a single discipline? My answer to these questions is simple --- it is the art of programming a computer. It is the art of designing efficient and elegant methods of getting a computer to solve problems, theoretical or practical, small or large, simple or complex.

C.A.R (Tony) Hoare
### Structuring Data

<table>
<thead>
<tr>
<th>“ant”</th>
<th>“bear”</th>
<th>“elephant”</th>
<th>...</th>
<th>“tiger”</th>
</tr>
</thead>
<tbody>
<tr>
<td>“elephant”</td>
<td>“bear”</td>
<td>“tiger”</td>
<td>...</td>
<td>“ant”</td>
</tr>
</tbody>
</table>

- **vector of strings/animals supporting operations:**
  - Find: is an word in the vector
  - Insert: a new word in the vector
  - Delete: a word from the vector

- **Which method is better?**
- **Can we be precise about how “hard” the operations are?**
Structuring Data II

- Consider lines of text (e.g., in an editor, book titles)
  - The dog ate the cat.
- add adjectives to the line of text
  - The small, brown dog ate the fat, furry cat.

- Structure a line to facilitate
  - efficient insertion/deletion of words
  - easy traversal of all the words in a line

- What’s the problem with using a vector here?
Problem solving: Finding Anagrams (Jumbles)

tangible, bleating
binary, brainy
aspired, despair, diapers, praised

- Given a list of words, find all the anagrams
- issues in problem statement? methods of solving?
Indirection: towards pointers

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;why&quot;</td>
<td>10</td>
<td>&quot;do&quot;</td>
<td>7</td>
<td>&quot;appear&quot;</td>
</tr>
</tbody>
</table>

- **sort**: alphabetically or by number of occurrences

```cpp
struct WordStat {
    string info;
    int count;
};

Vector<WordStat> list(100);

// swap elements of list
WordStat temp;
list[min] = list[j];
list[j] = temp;
```

- copying or moving data takes time, uses memory
- how much space to use in vector?
Vector<WordStat *> list(100);  // or list(100,0);

- Using a vector of pointers saves time and space
  - size of pointer is less than size of WordStat
  - less wasted space in vector
  - efficiency in copying/moving
- Create storage dynamically or at runtime, from the heap

```
list[0] = new WordStat;
list[0]->info = "ambitious";
(*list[0]).count = 1;
```
What is a Vector?

- **Encapsulation of an array (low-level structure)**
  - homogeneous aggregate, supports random access
  - declared in “vector.h”
  - templated class (what can’t go in a vector?)
  - based on standard STL class vector, but *safe*

- **Initialized to hold specific number of items**
  - who is responsible for ensuring there’s room?
  - what are alternatives? problems?

*Ask not what you can do to an object, but what an object can do to itself.*
What about anagrams? (see doana.cc)

- **Vector class has member functions** `append` and `size`
  - Work together when items added sequentially
  - Automatically resize vector (alternatives?)
- **How are two Anaword objects compared/printed?**
  - Use word form to print, e.g., bagel
  - Use *canonical* form to compare, e.g., abegl
- **What C++ features permit operators `<` and `==` to be used to compare Anaword objects?**
- **How is a program built from several classes/files compiled, linked, executed?**