Introduction to Informatics

Lecture 24:
Computing Models – More on Algorithms

"I like, you grab... Was that concept just a little too complex, Carl?"
Lecture notes
- Posted online
  - [http://informatics.indiana.edu/rocha/i101](http://informatics.indiana.edu/rocha/i101)
    - *The Nature of Information*
    - *Technology*
    - *Modeling the World*
- @ infoport
  - [http://infoport.blogspot.com](http://infoport.blogspot.com)
- From course package
    - Chapters 1, 4 (pages 1-12)
    - Chapter 10 (pages 13-17)
  - From Andy Clark’s book "*Natural-Born Cyborgs*"
    - Chapters 2 and 6 (pages 19 - 67)
  - From Irv Englander’s book "*The Architecture of Computer Hardware and Systems Software*"
    - Chapter 3: Data Formats (pp. 70-86)
    - Chapter 2: Classical Logic (pp. 87-97)
    - Chapter 3: Classical Set Theory (pp. 98-103)
    - Chapters 1-3 (pages 105-129)
    - OPTIONAL: Chapter 4 (pages 131-136)
    - Chapter 13 (pages 147-155)
    - Chapter 5 (pages 141-144)
  - Igor Aleksander, "Understanding Information Bit by Bit"
    - Pages 157-166
  - Ellen Ullman, "Dining with Robots"
    - Pages 167-172
Assignment Situation

Labs

Past

- Lab 1: Blogs
  - Closed (Friday, January 19): Grades Posted
- Lab 2: Basic HTML
  - Closed (Wednesday, January 31): Grades Posted
- Lab 3: Advanced HTML: Cascading Style Sheets
  - Closed (Friday, February 2): Grades Posted
- Lab 4: More HTML and CSS
  - Closed (Friday, February 9): Grades Posted
- Lab 5: Introduction to Operating Systems: Unix
  - Closed (Friday, February 16): Grades Posted
- Lab 6: More Unix and FTP
  - Closed (Friday, February 23): Grades Posted
- Lab 7: Logic Gates
  - Closed (Friday, March 9): Grades Posted
- Lab 8: Intro to Statistical Analysis using Excel
  - Closed (Friday, March 30): being graded
- Lab 9: Data analysis with Excel (linear regression)
  - Closed (Friday, April 6): Being Graded

Next: Lab 10

- Lab 10: Simple programming in Excel and Measuring Uncertainty
  - April 12 and 13, Due April 20

Assignments

Individual

- First installment
  - Closed: February 9: Grades Posted
- Second installment
  - Past: March 2: Grades Posted
- Third installment
  - Past: Grades Posted
- Fourth installment
  - Presented April 10th, Due April 20th

Group

- First installment
  - Past: March 9th, graded
- Second installment
  - Past: April 6th Being graded
- Third installment
  - Presented Thursday, April 12; Due Friday, April 27
Array of Integers

- A data structure to store series or lists or data
  - Example: age of students in I101
    - 19, 18, 21, 24, 19, 20, 19, 22, 18, 19
  - Index: stores the location of data element in the series
    - i: 1, 2, 3, 4, 5, 6, 7, 8, 9, ..., 100, ...
  - Array: stores data elements organized by index

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A:</td>
<td>19</td>
<td>18</td>
<td>21</td>
<td>24</td>
<td>19</td>
</tr>
<tr>
<td>i:</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>
Example: Sorting Algorithm

- **Insertion Sort**
  - Given a random sequence of numbers, sort them in increasing order
  - **Input**
    - \( S = <a_1, a_2, \ldots, a_n> \)
  - **Output**
    - A permutation or reordering of \( S: S'=<a'_1, a'_2, \ldots, a'_n> \), such that \( a_1 \leq a_2 \leq \ldots \leq a_n \)
    - Works the way many people sort a card hand
  - For \( j \leftarrow 2 \) to length \( S \) do
    - Key \( \leftarrow A[j] \)
    - \( i \leftarrow j-1 \)
    - While ((\( i > 0 \)) and (\( A[i] > \text{key} \))) do
      - \( i \leftarrow i-1 \)
    - endWhile
    - \( A[i+1] \leftarrow \text{key} \)
  - Endfor

\[ 6 \quad 2 \quad 3 \quad 1 \quad 5 \quad 4 \]
\[ 2 \quad 6 \quad 3 \quad 1 \quad 5 \quad 4 \]
\[ 2 \quad 3 \quad 6 \quad 1 \quad 5 \quad 4 \]
\[ 1 \quad 2 \quad 3 \quad 6 \quad 5 \quad 4 \]
\[ 1 \quad 2 \quad 3 \quad 5 \quad 6 \quad 4 \]
\[ 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \]
Insertion Sort Example

(a) 4 3 1 2
   4 1 2
   3
   4
   3

(b) 3 4 1 2
   3 4
   3
   1
   3
   4

(c) 1 3 4 2
   1
   3
   4
   1
   2
Individual Assignment – Part IV

- Step by step analysis of “dying” squares
  - 4th Installment
    - Presented: April 10th
    - Due: April 20th
- Use inductive and deductive reasoning
  - To uncover the algorithm in each quadrant
    - Build from inductive knowledge accumulated so far

Cycles = 1
Summary of Black Box

- **Quadrant 1**
  - At the random initial state
    - All numbers have equal probability of being initially present
    - But the probability of changes are different
  - In Any State
    - Any number changes depending on its neighbors
    - It ‘gravitates’ towards the smallest number that it ‘sees’ most often.
    - Odd and Even numbers do not show different behavior
- **What is the Algorithm?**
Summary of Black Box

- Quadrant 3
  - At the random initial state
    - All numbers have equal probability of being initially present
    - But the probability of changes are different
  - In Any State
    - 0 can only change to 0
    - 5 can only change to 5 or 0
    - Even digits always change to even digits
    - Odd digits could change to any other digit

- What is the Algorithm?

<table>
<thead>
<tr>
<th>n(i)</th>
<th>p(i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>27</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>12</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
</tr>
</tbody>
</table>

1. 0 → 0
2. \{5\} → \{0, 5\}
3. \{2, 4, 6, 8\} → \{0, 2, 4, 6, 8\}
4. \{1, 3, 7, 9\} → \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}
Summary of Black Box

- **Quadrant 2**
  - At the random initial state
    - All numbers have equal probability of being initially present
    - But the probability of changes are different
  - In Any State
    - 0 can only change to 0
    - 5 can only change to 5 or 0
    - Even digits always change to even digits
    - Odd digits could change to any other digit
- What is the Algorithm?
  1. 0 → 0
  2. {5} → {0, 5}
  3. {2, 4, 6, 8} → {0, 2, 4, 6, 8}
  4. {1, 3, 7, 9} → {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
# Possible Operations Q2 and Q3

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
<th>Excel</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>()</td>
<td>Brackets, grouping</td>
<td>()</td>
<td>y = (a + b) * (c + d)</td>
</tr>
<tr>
<td>*</td>
<td>Multiplication</td>
<td>*</td>
<td>i = j * k</td>
</tr>
<tr>
<td>+</td>
<td>Add</td>
<td>+</td>
<td>i = i + 1</td>
</tr>
<tr>
<td>-</td>
<td>Subtract</td>
<td>-</td>
<td>i = j - 3.2</td>
</tr>
<tr>
<td>/</td>
<td>Real division</td>
<td>/</td>
<td>i = 8 / 5 = 1.6</td>
</tr>
<tr>
<td>div</td>
<td>Integer division</td>
<td>Quotient (a,b)</td>
<td>i = 8 / 5 = 1</td>
</tr>
<tr>
<td>Mod, %</td>
<td>remainder</td>
<td>Mod (a, b)</td>
<td>i = 8 mod 5 = 3</td>
</tr>
<tr>
<td>ROUND</td>
<td>Rounds</td>
<td>ROUND (a, d)</td>
<td>i = ROUND(3.67, 0) = 4</td>
</tr>
<tr>
<td>INT</td>
<td>Integer Part</td>
<td>INT</td>
<td>i = INT(3.67) = 3</td>
</tr>
<tr>
<td>rand</td>
<td>Random number</td>
<td>Rand()</td>
<td>i = rand(n)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RandBetween(a,b)</td>
<td></td>
</tr>
</tbody>
</table>
Tip for Individual Assignment

- Quadrant Q
  - There are 100 cells in each 10x10 quadrant
  - $C = 1...100$
  - Each cell can take one of 10 colors
    - $V(C) = 0..9$
      - Is the value of the cell
      - This is the state cell $C$ is in
  - Random initialization of quadrant Q at cycle 1
    - For $c=1$ to $100$ do
      - $V(C) \leftarrow \text{randbetween}(0,9)$ \{random number 0 to 9\}
    - EndFor
    - Cycle $\leftarrow 1$
  - Run for Number of cycles
    - $n \leftarrow \text{Input dialog}$
    - For $k=1$ to $n$ do
      - Cycle $\leftarrow$ cycle+1
      - \{Pick random cell\}
      - $C \leftarrow \text{randbetween}(1,100)$
      - \{Update the value of the cell (NOT THE REAL THING)\}
      - $V(C) \leftarrow (\left( V(C) \times \text{randbetween}(0,9) \right) \text{ div 2}) - 5 \times X$
    - EndFor
  - X may be a hidden variable
    - $X \leftarrow ???$
In 1966 Joseph Weizenbaum developed an algorithm and program that simulates the behavior of a psychotherapist. The program seemed to be able to understand anything typed in by the user. The program was actually fairly “dumb” in modern AI terms. Its “understanding” was the result of programming trickery. Its weaknesses were caused by relying almost exclusively on the premise that the syntax of a sentence captured its semantic meaning.

Adapted from Bruce R. Maxim
Eliza Algorithm

- set up a language database
  - Words, synonyms, sentences
- begin the conversation (e.g. with a greeting)
- Repeat
  - read user input
  - generate Eliza's response
  - print the response on the screen
- until the conversation ends
Eliza Algorithm – More Details

- set up a language database
  - Words, synonyms, sentences
- begin the conversation (e.g. with a greeting)
- Repeat
  - read user input
    - Keeps track of the two most recent inputs from the user
  - generate Eliza’s response
    - preprocess the user input
      - Remove all punctuation from inputs and check for duplicate input
      - Make some synonym replacements from a list of pairs (e.g. big for huge)
      - Change pronouns (e.g. I and me to you)
    - find a matching keyword
    - choose an appropriate response template
      - if a keyword is found
        - extract the part of the user’s input following the keyword
        - apply transformations to the extracted input
        - plug the transformed input into the response
      - Else
        - generate a non-committal response
  - print the response on the screen
- until the conversation ends
Actroid

- The Actroid Robot
  - understands naturally spoken words and can carry on a conversation with a guest, answering in a natural voice.
    - Speaks Chinese, Korean, English and Japanese.
    - understands 40,000 phrases
    - 2,000 types of answers
    - Nuanced facial expressions, Natural gestures such as eye movement and smiling

- Robot information booth attendant at 2005 World Fair in Aichi
  - Built by Kokoro and Advanced Media
  - And she raps!
Next Class!

- **Topics**
  - Limits of Computation
  - Databases

- **Readings for Next week**
  - `@ infoport`
  - From course package
    - Igor Aleksander, "Understanding Information Bit by Bit"
      - Resources tab in onCourse.
    - Ellen Ullman, "Dining with Robots"
      - Resources tab in onCourse.

- **There is a lab this week!!!**
  - Lab 10
    - Simple programming in Excel and Measuring Uncertainty
      - April 13, 14; Due April 21