Introduction to Informatics
Lecture 22:
Computing Models - Algorithms

"I spent ten months reprogramming the Sims so they can talk... and the first thing they said was 'get a life!'"
NO LAB THIS WEEK !!!
Readings until now

- 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)
      - Due Friday, April 6
  - 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: Being Graded
  - Fourth installment
    - Presented April 10th, Due April 20th

- Group
  - First installment
    - Past: March 9th. Being graded
  - Second installment
    - March 29; Due Friday, April 6
Group Assignment

- Second Installment: Given the text of “Lottery of Babylon” by Jorge Luis Borges
  - Measures of central tendency and dispersion of letter frequency
  - Probability of a letter being a vowel
  - Probability of a letter being a consonant
  - Conditional probability of letters ‘e’ and ‘u’
    - $P(e|♥)$ where ♥ is the letter occurring before ‘e’
    - $P(u|♥)$ where ♥ is the letter occurring before ‘u’
    - Compute for all letters (not space)
    - Produce histogram of $P(e|♥)$, for all ♥.
    - Produce histogram of $P(u|♥)$, for all ♥.
    - Discuss the independence of ‘e’ and ‘u’ from other letters
  - Upload to Oncourse

\[
P(e | h) = \frac{|h \land e|}{|h|} = \frac{\text{he}'}{|h|}
\]

\[
P(e) = \frac{|e|}{N}
\]
Questions

- Over a 20-game period, the number of hits by a baseball player was
  - 1, 2, 0, 0, 1, 2, 2, 1, 0, 0, 4, 0, 1, 1, 3, 2, 1, 3, 0, and 1
  - Construct the Frequency distribution
  - In what proportion of games did he get at least 3 hits?
  - What is the mean, median, and mode
  - What is the line that best fits the data with the least squares criterion?

- A coin is tossed three times and an H or T (H= Head, T=Tail) is recorded each time.
  - List the elements of the sample space S and list the elements of the event consisting of
    - All heads
    - A head on the second toss
    - Two tails
  - Represent the sample space and the events above as a Venn Diagram

- One card is to be selected from an ordinary deck of 52 cards. Find the probability that
  - The selected card is an ace
  - The selected card is not a 9
Questions

- What type of Uncertainty does the Hartley measure of uncertainty measure?
- What are the units of Shannon entropy?
- Does Shannon’s information theory deal with the semantics and pragmatics of a message? Please explain why?
- If we have a symbol set \( X = \{A, B, C, D, E\} \) where the symbol occurrence frequencies are:
  - \( A = 0.5 \)
  - \( B = 0.2 \)
  - \( C = 0.1 \)
  - \( D = 0.1 \)
  - \( E = 0.1 \)
- If we know that a message is being sent in this language, what is the average minimum number of bits needed to guess the next symbol of the message?
Shannon’s entropy

**on average**, how many *yes-no* questions need to be asked to establish what the symbol is.

\[
H_S(A) = - \sum_{i=1}^{n} p(x_i) \log_2(p(x_i))
\]

\[
H_S \in [0, \log_2|X|]
\]

For one alternative

Uniform distribution
Entropy of an English Letter

- Entropy of English letter in a message
  - Uncertainty in guessing the next letter
  - Information contained in each new letter that arrives
- Assuming no word or sentence knowledge (no semantics)
  - From frequency distribution
    - $H_S(\text{letter})$ 4.18 bits
    - Hartley measure $= \log_2(26) = 4.7$ bits
  - How many guesses on average
- With knowledge of semantics
  - Tests with people in a sentence
    - $H_S(\text{letter})$ 1.1 bits
  - The value of semantics?

### Table

<table>
<thead>
<tr>
<th>Letter</th>
<th>Estimated Probability</th>
<th>log_2(p)</th>
<th>p_log_2(p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0.0917</td>
<td>3.61362011</td>
<td>0.29562469</td>
</tr>
<tr>
<td>b</td>
<td>0.0149</td>
<td>8.09884038</td>
<td>0.09042313</td>
</tr>
<tr>
<td>c</td>
<td>0.0278</td>
<td>6.18971731</td>
<td>0.14391844</td>
</tr>
<tr>
<td>d</td>
<td>0.0426</td>
<td>4.96839335</td>
<td>0.19394872</td>
</tr>
<tr>
<td>e</td>
<td>0.127</td>
<td>2.97709988</td>
<td>0.37809186</td>
</tr>
<tr>
<td>f</td>
<td>0.0223</td>
<td>6.48881248</td>
<td>0.12235682</td>
</tr>
<tr>
<td>g</td>
<td>0.0202</td>
<td>6.82860099</td>
<td>0.11371692</td>
</tr>
<tr>
<td>h</td>
<td>0.0909</td>
<td>4.03741398</td>
<td>0.24668286</td>
</tr>
<tr>
<td>i</td>
<td>0.0897</td>
<td>3.84287363</td>
<td>0.26739022</td>
</tr>
<tr>
<td>j</td>
<td>0.0016</td>
<td>9.39821781</td>
<td>0.01407123</td>
</tr>
<tr>
<td>k</td>
<td>0.0077</td>
<td>7.02092684</td>
<td>0.05408113</td>
</tr>
<tr>
<td>l</td>
<td>0.0403</td>
<td>4.83397835</td>
<td>0.16971298</td>
</tr>
<tr>
<td>m</td>
<td>0.0241</td>
<td>6.38472304</td>
<td>0.12863324</td>
</tr>
<tr>
<td>n</td>
<td>0.0876</td>
<td>3.88688889</td>
<td>0.26520539</td>
</tr>
<tr>
<td>o</td>
<td>0.0761</td>
<td>3.73604328</td>
<td>0.29060176</td>
</tr>
<tr>
<td>p</td>
<td>0.0193</td>
<td>6.89628534</td>
<td>0.10891843</td>
</tr>
<tr>
<td>q</td>
<td>0.0011</td>
<td>9.98678428</td>
<td>0.00996798</td>
</tr>
<tr>
<td>r</td>
<td>0.0099</td>
<td>4.08159019</td>
<td>0.24327188</td>
</tr>
<tr>
<td>s</td>
<td>0.0333</td>
<td>3.98168089</td>
<td>0.26033949</td>
</tr>
<tr>
<td>t</td>
<td>0.0508</td>
<td>3.46346141</td>
<td>0.31369887</td>
</tr>
<tr>
<td>u</td>
<td>0.0276</td>
<td>6.17916972</td>
<td>0.14245669</td>
</tr>
<tr>
<td>v</td>
<td>0.0081</td>
<td>8.67300264</td>
<td>0.08859542</td>
</tr>
<tr>
<td>w</td>
<td>0.0238</td>
<td>6.40588333</td>
<td>0.12768984</td>
</tr>
<tr>
<td>x</td>
<td>0.0016</td>
<td>9.39821781</td>
<td>0.01407123</td>
</tr>
<tr>
<td>y</td>
<td>0.0167</td>
<td>6.88888688</td>
<td>0.11161351</td>
</tr>
<tr>
<td>z</td>
<td>0.0007</td>
<td>10.48135768</td>
<td>0.00739826</td>
</tr>
</tbody>
</table>

$$H_S(A) = - \sum_{i=1}^{n} p(x_i) \log_2(p(x_i))$$
The Modeling Relation

Hertz’ Modeling Paradigm

- Modeling
  - Compute hypothesis
- Rules from Inductive and Deductive Analysis
  - From Data analysis
  - Produce Conclusions
What's an Algorithm?

Duracell: The official battery of Al Gore

Al Gore Rhythm?
Algorithms

- **OED**
  - **Math**: A process, or set of rules, usually one expressed in algebraic notation, now used especially in computing, machine translation and linguistics.

- **Medicine**: A step-by-step procedure for reaching a clinical decision or diagnosis, often set out in the form of a flow chart, in which the answer to each question determines the next question to be asked.

- **Specifically**
  - A set of instructions or procedures **for solving a problem**
  - For calculating or **computing a model**.
Algorithms are like Recipes

Recipe
CHOCOLATE CAKE

4 oz. chocolate 3 eggs
1 cup butter 1 tsp vanilla
2 cups sugar 1 cup flour

Melt chocolate and butter. Stir sugar into melted chocolate. Stir in eggs and vanilla. Mix in flour.
Spread mix in greased pan. Bake at 350 for 40 minutes or until inserted fork comes out almost clean. Cool in pan before eating.

Program Code

Declare variables:
chocolate  eggs  mix
butter   vanilla
sugar    flour

mix = melted ((4 oz chocolate) + butter)
mix = stir (mix + (2 sugar))
mix = stir (mix + (3 eggs) + vanilla)
mix = mix + flour
spread (mix)
While not clean (fork)
bake (mix, 350)
Algorithm

- Term derived from the name of the Persian mathematician Al-Khwarizmi
  - Lived in the VIII or IX century AD in Baghdad
  - Derived the concept
- In Computer Science
  - A well-defined computational procedure that takes some input values and produces output values, in a finite amount of time using a finite set of well-defined operations
  - To solve a computational problem
    - A desired input/output relationship
Example: Sorting

Problem:
- 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 \( a_1 \leq a_2 \leq \ldots \leq a_n \)

Instance of the problem
- I: \( <89, 54, 7, 102, 73, 15> \)
- O: \( <7, 15, 54, 73, 89, 102> \)

Correct Algorithm
- If for every input instance, it halts with the correct output
- A correct algorithm solves the computational problem
Pseudocode

- It is important that algorithms are unambiguous and precise as possible.
  - Conventions to attain layout and terminology.
- Algorithms often divided into sections
  - Input
    - the parts/components/ingredients required to accomplish the task
  - Processing
    - Actions/steps/methods to produce a result
  - Output
    - the required outcome
- Pseudo Code
  - Fake code, not really programming code
  - Specifies the steps required for processing.
  - Structured language used to specify an algorithm.

“It has often been said that a person does not really understand something until he teaches it to someone else. Actually a person does not really understand something until he can teach it to a computer.” Donald Knuth

author of the *The Art of Computer Programming*, father of the field of rigorous analysis of algorithms, creator of the TEX typesetting system, etc…
Advantage of pseudocode

- **Reduced complexity**
  - While writing the algorithm the developer can focus on solving the problem, not how it is written in a particular language.

- **Increased flexibility**
  - Pseudo code is written so that code based on it should be able to be written in any language.

- **Ease of understanding**
  - No need to understand a particular programming language, more like natural language.
  - Employs whatever expressive method is most clear and concise.
    - Even a plain English sentence.
Pseudocode Statements

- **Assignment**
  - Used to (a) store a value in a *variable* or (b) calculating the answer to an arithmetic problem and then storing the result.
  - **Symbols used**
    - "=" or "←"
  - **Example**
    - Total = 100 (storing a value)
    - Area = Length * Width (arithmetic Calculations)

<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>
Pseudocode I/O

- **Input**
  - Display a message asking the user for a value and store the value typed by the user in a variable.
  - **Examples**
    - *Input custNam*
      - displays a message asking the user to input a customers name and store the value typed by the user in the *variable* called `custName`.

- **Display/Output**
  - Displays data on the computer screen (monitor).
  - **Examples**
    - *Display "Width = ", width*
    - *Display "Hello World"*
    - *Display grossIncome, taxPayable*
      - Values in quotation marks are displayed exactly as stated (minus the quotation marks)
      - The values held in variables are displayed rather than the variable name.
Pseudocode Decision

- **If-then-else**
  - If (letter = “a” or letter = “A”) then
    - display “1”
    - Count_A = Count_A +1
  - Else
    - Display “0”
  - End-if

- **Case**
  - Case letter of
    - “a” or “A”: display “1”, count_a = count_a+1
    - “b” or “B”: display “2”, count_b = count_b+1
    - “c” or “C”: display “3”, count_c = count_c+1
  - Else
    - Display “0”
  - End-case
Pseudocode Iteration or Loops

- **For**
  - **For** \(x = 1\) to 100 **do**
    - \(y = \text{rand}(100) \mod x\)
    - Display \(y\)
  - **ENDFOR**
    - Specifies exactly how many iterations to compute

- **While**
  - \(x = 1\)
  - **While** \(((y \leq 4) \text{ and } (x \leq 100))\) **do**
    - \(y = \text{rand}(100) \mod x\)
    - Display \(y\)
    - \(X = x + 1\)
  - **ENDWHILE**
    - The number of iterations to compute may depend on the computation itself
Example: Hire assistant

```
HIRE-ASSISTANT(n)
1  best ← 0  \(\triangleright\) candidate 0 is a least-qualified dummy candidate
2  for i ← 1 to n
3      do interview candidate i
4      \[\text{if candidate } i \text{ is better than candidate } best\]
5      \[\text{then } best ← i\]
6
```

```
RANDOMIZED-HIRE-ASSISTANT(n)
1  randomly permute the list of candidates
2  best ← 0  \(\triangleright\) candidate 0 is a least-qualified dummy candidate
3  for i ← 1 to n
4      do interview candidate i
5      \[\text{if candidate } i \text{ is better than candidate } best\]
6      \[\text{then } best ← i\]
7
```

Hire Candidate Best
Flow Chart

- Pictorial representation of algorithm
  - Parallelogram for input/output
  - Oval for start and stop
  - Rectangle for processing
  - Diamond for decision
  - Hexagon for preparations and loops
  - Circle for connector
  - Arrow for flow direction
Flowchart Examples

GEEN DATING FLOWCHART

START

DO YOU HAVE SOMEONE IN MIND?

YES NO

ARE THEY AVAILABLE?

YES NO

ASK THEM IF THEY LIKE THE HITCHHIKER'S GUIDE:

YES NO

ASK THEM IF THEY LIKE STAR WARS:

YES NO

ASK THEM IF THEY LIKE COMPUTERS:

YES NO

WHAT DO THEY KNOW YOU EXIST?

YES NO

TRULY REMARKABLE, THEY'RE AVAILABLE AND THEY KNOW YOU EXIST, WILL WONDERS NEVER CEASE.

YES NO

SO NOW YOU JUST NEED TO ASK THEM OUT ON A DATE, WHAT DO THEY SAY?

YES NO

YEAH RIGHT!

YEAH RIGHT!

YEAH RIGHT!

YEAH RIGHT!

THAT'S ENOUGH.

NO

WHAT ABOUT PRAYING?

NO

YEAH RIGHT!

RAH RAH RAH...

RAH RAH...

RAH RAH...

RAH RAH...

STOP

ARRIVE SAFELY

LEAVING THE OFFICE

CHECK THE TIME AND WEATHER

WEATHER CLEAR?

YES NO

BEFORE 5:00PM?

YES NO

CHECK FOR CONGESTION ON PRIMARY ROUTE

PRIMARY CONGESTED?

YES NO

TAKE ALTERNATE "A" HOME

DIVERT TO ALTERNATE "B"

TAKE THE PRIMARY ROUTE HOME

http://cset.sp.utoledo.edu/sample/engt1050/engt1050_flowchart.html

Copyright © 2002 Lido
Next Class!

- **Topics**
  - More Algorithms and Limits of Computation

- **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.

- No lab this week!!!!!!