About This Presentation

This presentation belongs to the lecture series entitled “Ten Puzzling Problems in Computer Engineering,” devised for a ten-week, one-unit, freshman seminar course by Behrooz Parhami, Professor of Computer Engineering at University of California, Santa Barbara. The material can be used freely in teaching and other educational settings. Unauthorized uses, including any use for financial gain, are prohibited. © Behrooz Parhami

<table>
<thead>
<tr>
<th>Edition</th>
<th>Released</th>
<th>Revised</th>
<th>Revised</th>
<th>Revised</th>
<th>Revised</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>May 2012</td>
<td>May 2015</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Word Search Puzzles
Type 1, With Word List Supplied

The puzzle below is a little harder than the normal word search: one of the 36 first/last names has been left out (which one?)

AGITATOR
ASSEMBLY
CLUTCH
CONNECTORS
CONTROL
COUPLING

GLIDE
LINT SCREEN
PULLEY
SEAL
SWITCH
VALVE

YOUR MO J D ZER NAS O
SUSAN ARABRA BHTR
RTVBEHNYYWDLOG
UKCALRIALBKRNJL
FSLHIGYLMEPaulA
MPASYOMVMAHGI
AMCNRUNIAYPCULS
HKAOABNETSRIKAZ
GLTCFGGOSSRTRVPE
NUEAUVEOESAEAR
IFLERBRZBR nec LN
BPLBEKIMMEL LILE
NOSLENO JER BAKER

AMY STEEL
KEVIN BLAIR
RON PALILLO
BARBARA BINGHAM
KIRSTEN BAKER
SHAVAR ROSS
BRUCE MAHLER
LARRY ZERNER
STU CHARNO

CAROL LACATELL
MARK NELSON
SUSAN BLU
DANA KIMMELL
PAUL KRATKA
TONY GOLDWYN
JOHN FUREY
RICHARD YOUNG
TRACIE SAVAGE

May 2015

String Matching

Slide 3
“Ten Puzzling Problems in Computer Engineering” Word Search

WORD LIST:

BINARY SEARCH
BYZANTINE GENERALS
CRYPTOGRAPHY
EASY HARD IMPOSSIBLE
MALFUNCTION DIAGNOSIS
PLACEMENT AND ROUTING
SATISFIABILITY
SORTING NETWORKS
STRING MATCHING
TASK SCHEDULING

Puzzle generated at:
http://puzzlemaker.school.discovery.com/WordSearchSetupForm.html
Word Search Puzzle
Type 2, With Clues Supplied for the Words to be Found

Seven birds
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

Five units of length
☐ ☐ ☐ ☐ ☐

Four currencies
☐ ☐ ☐ ☐

Two things football players wear
☐ ☐

Large gland in the neck
☐

USA Today’s “Word Roundup” for May 16, 2007: http://puzzles.usatoday.com/
A Challenging Hybrid Word Search Puzzle
Word Search Puzzle with a Twist

This “Missing Peace Puzzle” was used in a qualifying round of World Puzzle Championships. Supply the 16 missing letters at the center of the grid so that the word-search puzzle contains 18 of the 19 names of Nobel Peace Prize winners listed.

ARIAS
BRANDT
BRIAND
DUNANT
HULL
HUME
KING
MANDELA
MARSHALL
MONETA
MOTT
PAULING
PEARSON
PERES
RABIN
RENAULT
SADAT
SAKHAROV
WALESA
String Matching: Problem Definition

Given a data string with \( n \) symbols and a pattern string with \( m \) symbols:

1. Does the pattern string appear in the data string?
2. What are the locations of all occurrences of the pattern in the data?

The brute-force, or sliding window, algorithm

Consider all possible positions where the pattern might begin \((n - m + 1)\)
For each start position, do up to \( m \) comparison to see if there is a match

Worst-case complexity = \( O(mn) \); e.g., pattern “aaaaa”, data “aaaaaaaaaaaa”
## Converting 2D Search Puzzles to 1D Searches

A 2D word search puzzle looks more exotic but it can be readily converted to a 1D string search puzzle.

### Row-major order

Insert a special symbol (\#) between rows to ensure that new words or patterns are not created by the expansion.

### Column-major order

Similarly for (anti)diagonal positions.

---

May 2015

String Matching

BParham
Finding a Needle in a Haystack

Search for the 10-symbol “needle” h-e-l-e-n- -h-u-n-t in the Internet “haystack” with many TBs of data

The brute force algorithm amounts to: “Look in this corner, now in this other corner, then over there, and so on.”

The Internet holds some $1^\dagger$ trillion pages, growing by billions a day; each page on average contains in excess of 10 KB, say

$m \approx 10$

$n \approx 10^{12} \times 10^4 = 10^{16}$ B = 10 PB (Petabyte) = 0.01 EB (Exabyte)

$O(mn) \approx 10^{17}$ comparisons $\Rightarrow 10^8$ s (> 3 years), with $10^9$ comparisons/s
Needle in a Haystack: Internet Search

Search for the 10-symbol string “helen hunt”

2.1M hits 3 years ago
5.4+M hits in mid 2012
Needle in a Haystack: Doing Less Work

For a particular pattern and unpredictable data strings, preprocess the pattern so that searching for it in various data strings becomes faster

Analogy: Magnetize the needle

For a particular data string and unpredictable patterns, preprocess the data string so that when a pattern is supplied, we can readily find it with much less work

Analogy: Do a thorough search of the haystack for different types of needles and place markers to guide future searches
Example of Preprocessing the Pattern String

Devise an efficient method for finding the pattern “abcbab” in various data strings formed from the symbols a, b and c.
**Example of Preprocessing the Data String**

Devise an efficient method for finding various patterns in the data string:

```
|   | a | b | c | b | b | b | a | b | c | b | a | b | b | c | a | b | c | b | a | b | c | b | a | b | c | b | b |
| 0 | a | a | b |   | 14 |
| 1 | a | b | b |   | 10 |
| 2 | a | b | c |   | 0, 6, 15, 19, 23 |
| 3 | b | a | b |   | 5, 9, 18, 22 |
| 4 | b | b | a |   | 4 |
| 5 | b | b | b |   | 3 |
| 6 | b | b | c |   | 11 |
| 7 | b | c | a |   | 12 |
| 8 | b | c | b |   | 1, 7, 16, 20, 24 |
| 9 | c | a | a |   | 13 |
|10 | c | b | a |   | 8, 17, 21 |
|11 | c | b | b |   | 2, 25 |
```

Find all occurrences of the pattern “```abcbab```”

```
<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td></td>
<td>0, 6, 15, 19, 23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>b</td>
<td>c</td>
<td>b</td>
<td></td>
<td>1, 7, 16, 20, 24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>b</td>
<td>c</td>
<td>b</td>
<td></td>
<td>8, 17, 21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>b</td>
<td>a</td>
<td>b</td>
<td></td>
<td>5, 9, 18, 22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Alternate strategy: Focus on the locations of ```abc``` and ```bab```
Search Engine Indexes

17.5B hits 3 years ago
25.3B hits in mid 2012

667 hits a few years ago
1M+ hits in mid 2012
Approximate String Matching

Notion of string distance
Each of the following transformations in a string creates a distance of 1
1. Insertion of an extra symbol
2. Deletion of a symbol
3. Transposition of two adjacent symbols

Example distance-1 strings for *helen hunt*:
- hellen hunt (Insertion)
- elen hunt (Deletion)
- helen hnut (Transposition)

Example distance-2 strings for *helen hunt*:
- hellen hnut (Insertion + Transposition)
- elen huntt (Deletion + Insertion)
- lheen hunt (2 Transpositions)

Wildcard symbols can help in formulating approximate string searches
*h* *hunt* means any string that begins with an “h”, ends with “hunt”, and
has an arbitrary set of symbols between the two

Melvyl (UC library catalog) allows such searches, e.g., author: *hunt h*
The (DNA) Sequence Alignment Problem

Given sequences S1 and S2 composed of the letters A, C, G, T
Determine their degree of similarity

S1: A G G G C T
S2: A G G C A

Application: Matching a given DNA sequence to a set of sequences in a database to find the best match

Dissimilarity arises from missing letters or mismatched letters

Alignment 1:

Penalty = GC mismatch + CA mismatch + 1 gap

Alignment 2:

Penalty = 1 gap + TA mismatch

Optimal alignment found via dynamic programming