CS 170: Algorithms

Lecture in a minute.

Fast Fourier Transform.


A(x) = A_0(x^2) + x(A_0(x^2))

where
A_0(x) = a_0 + a_1x + a_2x^2 + \ldots

Even coefficient polynomial:
A_0(y) = a_0 + a_1y + a_2y^2 + \ldots

Odd coefficient polynomial:
A_0(y) = a_1 + a_2y + a_3y^2 + \ldots

Proof:
Should get a_dx^d in A_0(x^2) + xA_0(x^2).
d is even:
d/2th coefficient of A_0(y) is a_{d/2}.
When y = x^2, a_{d/2}x^{d/2} = a_{d/2}(x^2)^{d/2} = a_dx^d

d is odd:
i = \frac{d-1}{2} in coefficient of A_0(y) is a_{d/2}.
When y = x^2, A_0(y) contains a_{d/2}x^{(d-1)/2} = a_{d/2}(x^{d-1})

Multiplying Complex Numbers

Complex numbers:
Coordinate representation: a + bi
Polar representation: (r, \theta) where r = \sqrt{a^2 + b^2} and \theta = \tan^{-1} \frac{b}{a}

(\rho, \theta_1) \times (\rho_2, \theta_2) = (\rho \rho_2, \theta_1 + \theta_2)

The n-th complex roots of unity.

Solutions to \ z^n = 1
(1, \frac{2\pi}{n})^n = (1, \frac{2\pi \times n}{n}) = (1, 2\pi k) = 1

Pair w/common square!

Squares: n-th roots.
1. Graphs
2. Reachability.

Scheduling: coloring.

Directed acyclic graphs.

Chemical networks.

Graph Implementations.

Matrix Representation.

Adjacency List

Edge \((u,v)\)? \(O(1)\) \(O(|V|)\)

Neighbors of \(u\) \(O(|V|)\) \(O(d)\)

Space \(O(|V|^2)\) \(O(|E|)\)

Exam Slot 1.
Exam Slot 2.
Exam Slot 3.

Fewer Colors?
Yes! Three colors.

Fewer Colors?
Four colors required!
Theorem: Four colors enough.

From http://www.graphviz.org/content/crazy.

Test your understanding.

Adjacency list of node 0?
(A) 0 : 1
(B) 0 : 1, 2
(C) 0 : 2
(C)
How many edges?
(A) 2
Total length of adacency lists?
(A) 2
(B) 3
(C) 4
(C) 2 entries for each edge!

Exploring a maze.

Theseus: gotta kill the minatour in the maze
Ariadne: he’s cute...fortunately she’s smart.

Gives Theseus Ball of Thread and Chalk!
Explore a room:
Mark room with chalk.
For each exit:
- Look through exit. If marked, next exit.
- Otherwise go in room unwind thread.
Explore that room.
Wind thread to go back to “previous” room.

Searching

Find a minatour!
Find out which nodes are reachable from A.

Correctness.

Explore(v):
1. Set visited[v] := true
2. For each edge (v,w) in E
3. if not visited[w]: Explore(w).

Property:
All and only nodes reachable from A are reached by explore.

Only: when u visited, stack contains nodes in a path from a to u.
All: if a node u is reachable, there is a path to it. Assume: u not found.

z is explored, w is not! Explore (z) would explore(w)! Contradiction.
Proof was induction.

Property: Every node with a path of length \( k \) or less is reached.

Induction by Contradiction.
Find smallest \( k \) (path length) where property doesn't hold.
It does hold.
No smallest \( k \) where it fails.
Must hold for every \( k \).

Done!!! or

Running Time.

Explore(v):
1. Set visited\([v]\) := true.
2. For each edge (v,w) in E
3. if not visited\([w]\): Explore(w).

How to analyse?
Let \( n = |V| \), and \( m = |E| \).

\[ T(n,m) \leq (d)T(n-1,m) + O(d) \]

Exponential ?!?!?!

Don't use recurrence!

Lecture in a minute.

FFT Wrapup:
Evaluate degree \( n \) polynomial on \( n \) points.
Recursion:
Evaluate Odd/Even polynomials on squares of points.
Which \( n = 2^k \) points?
3rd root of unity: complex numbers.

“squares of \( n \)th root of unity are \( n/2 \)th roots of unity.”
Evaluate Odd/Even polynomials on \( n/2 \) points.

\[ T(n) = 2T(n/2) + O(n) = O(n\log n). \]

Graphs:
\( G = (V,E) \), \( V \) - vertices, \( E \) edges.

Representations:
Matrix: \( |V|^2 \) space, fast check for edges.
Adjacency List: \( O(|V| + |E|) \) space. More complicated.

Procedure: explore(v).
Explores the graph.
Uses a stack.
Nonrecursive non-loop counting Runtime analysis.

Every little move she makes...