

Advanced Algorithms

Lecture 1

Lecture by Shay Mozes
Typeset by Steven Karas

2016-11-08
Last edited 17:26:16 2016-11-28

Recommended Reading:

- Introduction to Algorithms, MIT Press (commonly referred to as CLRS)
- Others will be mentioned in the slides

Lecture Videos:

- [Videos from 2014](#)
- [Videos from 2015](#)
- [Videos from 2016](#)

1 Algorithm Modelling

Begin by formalizing the goal. From there we can see what sort of requirements we have, and what restrictions are present (online, streaming, time/size restrictions, approximations, etc). Check literature and references for known algorithms/solutions, and then adapt to our current problem.

1.1 Algorithm Design Goals

1. Correctness
2. Efficiency (time/space/wall time)
3. Simplicity
4. Ease of implementation

1.2 Evaluating Algorithms

Efficiency is measured using a complexity function: $T(n) : \mathbb{N} \rightarrow \mathbb{N}$ where $T(n)$ is the number of operations the algorithm does on an input of size n

We can measure this in several ways:

1. Worst case
2. Best case
3. Average case

1.3 RAM Model of Computation

Each C statement takes 1 time step. Loops and calls are not simple ops. Each memory access takes one time step, and no limitation on memory. For any given problem, the running time is the number of simple operations, and the space is the number of memory cells touched.

2 Big O Notation

$f(n) = O(g(n))$ if $c \cdot g(n)$ is upper bound on $f(n)$ iff there exist c, N such that for any $n \geq N$, $f(n) \leq c \cdot g(n)$

Common complexity classes:

- $O(n)$
- $O(\log n)$
- $O(2^n)$
- $O(n^c)$
- P
- NP
- $PSPACE$

In the homework, we'll probably see $O(2^{\sqrt{\log n}})$

3 Graph theory primer

A graph is a 2-tuple of vertices and edges: $G = (V, E)$ where $|V| = n$, $|E| = m$ can have weights (as a $c(e) : E \rightarrow \mathbb{R}$, can be directed (meaning that an edge has a source and target), can even have multiple edges per vertex pair.

It can be useful to assume there are no cycles in a graph. It's so commonly used, we even call a directed acyclic graph a DAG for short.

4 Graph search

- DFS (uses a stack)
- BFS (uses a queue)

For an extra challenge, try changing the BFS on slide 16 to use a stack instead and find a counterexample of a graph for which it does NOT work correctly.

5 Shortest Path

5.1 Single Source

Single source: given a source vertex, find the shortest path to any other vertex of G

BFS unweighted

Dijkstra only for non-negative weights. $O(n^2)$ or $O(m+n\log n)$ with fibonacci heaps; $O((m+n)n\log n)$ for binary heaps

Bellman Ford arbitrary weights. $O(nm)$

5.2 All pair

Given a graph, find the shortest path between all pairs of vertices.

Floyd Warshall $O(n^3)$

Johnson $O(mn + n^2\log n)$

Williams $O(\frac{n^3}{2^{\sqrt{\log n}}})$

6 Minimum Spanning Tree

Given a weighted graph, find subset of E that minimizes $\sum C(e \in E)$

Kruskal Greedily add lightest edge that doesn't form a cycle. $O(m \log m)$ using basic sorting and Union-Find data structures.

Prim Grow a tree by greedily adding the lightest edge with one endpoint in the tree and one not in the tree. $O(m \log m)$ using binary heap. $O(m + n \log n)$ using Fibonacci heaps.

Boruvka $O(m \log m)$

7 Max flow

Given a DAWG, source and sink, find max flow of network.

st-Cut a subset of E that partitions the graph into non-empty sets S and T where $s \in S$ and $t \in T$. The maximum flow of the network is the minimum of all the st-cut flows.

8 Induction

1. Loop invariants - expression that remains true for all iterations of a loop
2. Recursion - self-referential function
3. Recurrences - See lecture video for explanation

Given a predicate P , $P(k)$ for some k (usually $k = 0$). $P(n) \rightarrow P(n+1)$ for all $n \geq k$; $P(n) = P(k)$ for all $n \geq k$.

8.1 Example: Geometric Sequence

$$\forall a \neq 1$$

$$a^0 + a^1 + \dots + a^n = \frac{a^{n+1} - 1}{a - 1}$$

$$\text{Base: } n = 0; a^0 = \frac{a^{0+1} - 1}{a - 1}$$

$$a^0 = 1 = \frac{a^1 - 1}{a - 1}$$

$$\text{Assume: } a^0 + a^1 + \dots + a^n = \frac{a^{n+1} - 1}{a - 1}$$

$$\begin{aligned} a^0 + a^1 + \dots + a^{n+1} &= a^0 + a^1 + \dots + a^n + a^{n+1} \\ &= \frac{a^{n+1} - 1}{a - 1} + a^{n+1} = \frac{a^{n+1+1} - 1}{a - 1} \end{aligned}$$

8.2 Example: Tiling a floor

Tiling problem: given a room of size 2^n by 2^n , tile the room using L-shaped tiles covering 3 units. Also reserve a single tile for a "Logo".

Better explained in the video than I ever could.

9 Matching problems

Given a set of boys and a set of girls, each with a ranking of all the other sex.

9.1 What's a good pairing?

- Maximize total satisfaction
- Maximize minimum satisfaction
- Maximize maximum difference
- Maximize number of people with first choices
- No stable pairings - all reasonable solutions must have stability

9.2 The roommate problem

Given a set of freshman of size $2n$, each with a ranking for who will be their roommate.

Such that:

Table 1: Roommate preferences

Adam	C	B	D
Bob	A	C	D
Charlie	B	A	D
Dave	A	B	C

With an odd number of cycles, and a single choice at the bottom of the list for everyone else, then there will always be an unstable couple.

9.3 Traditional marriage algorithm

1. Each morning, boys go to top girl in their list.
2. Girl selects highest boy in her list, rejects all others.
3. Boys remove rejections from list.
4. Terminate if all boys were selected on same day.

9.4 Trade up lemma

If on day i a girl selects a boy b , she is guaranteed to marry someone she likes at least as much as b .

$\forall k \geq 0$, on day $i + k$ the girl will select a boy she likes as much as b .

Base $k = 0$: trivially true.

Assume condition is true for $k - 1$. Thus she has a boy at least as good as b in the running for day $i + k$.

See formal induction proof on slide 62

Corollary Each girl will marry her absolute favorite among the boys who visited her during TMA.

9.5 Does TMA always terminate?

Either it doesn't specify a next step (i.e. whether the algorithm is well defined), or it might keep going for an infinite number of days.

Proof Each boy can be rejected at most $n - 1$ times. This would imply that there are n couples. Assume that Bob is rejected by all the girls, and that some woman Amy, has no proposers when Bob was rejected the last time. By the trade up lemma, Amy was never proposed to, but Bob proposed to everyone, since he was unmatched. In particular, Amy. Therefore, Bob must **not** have been rejected by all the girls.

Consider a list containing all the boy's preferences of the girls. There are n boys, and each list has n girls in it, giving us a total of n^2 girls names in the list. Each day, there is at least one boy who gets rejected, so at least one girl gets crossed off this list. Therefore, the number of days is bound by the original size of the list.

Now that we've proven it always terminates, is it stable? We'll use the trade up lemma as the definition of stable (proof by contradiction):

Given two couples, Alice and Bob, Claire and Dave. Now suppose that Bob prefers Claire over Alice, and Claire prefers Bob over Dave (aka a rogue couple). By this, Bob must have visited Claire earlier than Alice, which would imply that Claire rejected him. As such, this contradicts the trade-up lemma, because Claire chose Luke over Bob. Therefore, Bob and Claire cannot be a rogue couple.

9.6 Optimality

Is this optimal for everyone, or does it prefer one sex?

optimal: highest ranking spouse they could receive in a stable matching.

pessimal: lowest ranking spouse they could receive in a stable matching.

A matching is male-optimal if every boy receives his optimal spouse. A matching is male-pessimal if every boy receives his pessimal spouse.

TMA is male-optimal and female-pessimal.

Proof Suppose that some boy gets rejected by his optimal girl in TMA. For example, given Bob is the first boy rejected by his optimal girl Mia. She selected Luke instead. This leads to Bob and Mia are an unstable couple, contradicting the existence of a stable match where Mia is the optimal spouse for Bob.

Given a stable match S where Alice does worse than in T . Let Luke be her spouse in T . Alice likes Luke better than her spouse in S . T is male-optimal, so Luke likes Alice better than his spouse in S . Therefore, S is not stable, and as such, T is female-pessimal.

We did not discuss implementation in $O(n^2)$.