

Objectives

- Finish implementation of Stable Matching
 - Get out your handouts
- Survey of common running times

Turn in completed problem sets

Review

- What does $O(f(n))$ mean?
 - Intuitive
 - More precise definition
- What are the other bounds we discussed?

A Fashion Analogy

- O == Hammer pants
 - Loose and baggy with plenty of room for the pants to shrink or the body to grow
- Ω == The pants you plan to fit in this summer after working off the snacks from Christmas
- Θ == Katy Perry's skin tight jeans in a teenage dream
 - Can't make them any smaller, and no extra room to even fit a cell phone in the pocket

Courtesy Andy Danner, Swarthmore

Jan 19, 2018

Sprenkle - CSCI211

3

Review: Gale-Shapley Stable Matching Algorithm

```

Initialize each person to be free
while (some man is free and hasn't proposed to every woman)
  Choose such a man m
  w = 1st woman on m's list to whom m has not yet proposed
  if (w is free)
    assign m and w to be engaged
  else if (w prefers m to her fiancé m')
    assign m and w to be engaged and m' to be free
  else
    w rejects m
  
```

Jan 19, 2018

Sprenkle - CSCI211

4

Stable Matching Implementation

- What do we need to represent?
- How should we represent them?

Data	How represented
Men, Women	
Preference lists	
Unmatched men	
Who men proposed to	
Engagements	

What's the difference between an array and a list?

Jan 19, 2018

Sprenkle - CSCI211

5

Arrays



- *Fixed* number of elements
- What is the runtime of
 - Determining the value of the i^{th} item in the array?
 - Determining if a value e is in the array?
 - Determining if a value e is in the array if the array is sorted?

Jan 19, 2018

Sprenkle - CSCI211

6

Array Operations' Running Times

Operation	Running Time
Value of i^{th} item	$O(1) \rightarrow$ direct access
If e is in the array	$O(n) \rightarrow$ look through all the elements
If e is in the array if sorted	$O(\log n) \rightarrow$ binary search

Limitation of arrays?

Fixed size $\rightarrow$ difficult to add/delete elements

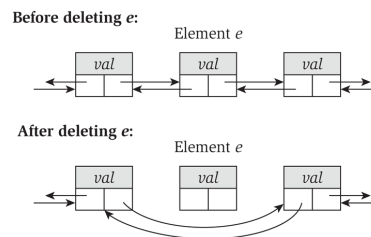
Jan 19, 2018

Sprenkle - CSCI211

7

Lists

- Dynamic set of elements
 - Linked list
 - Doubly linked list
- What is the running time to
 - Add an element to the list?
 - Delete an element from the list?
 - Find an element e in the list?
 - Find the i^{th} element in the list?



Jan 19, 2018

Sprenkle - CSCI211

8

List Operations' Running Time

Operation	Running Time
Add element	$O(1)$
Delete element	$O(1)$
Find element	$O(n)$
Find i^{th} element	$O(i)$

Disadvantage of list instead of array?

Finding i^{th} element is slower

Jan 19, 2018

Sprenkle - CSCI211

9

Converting between Lists and Arrays (and Vice Versa)

- What is the running time of converting a list to an array?
- An array to a list?

$O(n)$

Jan 19, 2018

Sprenkle - CSCI211

10

Stable Matching Implementation

- What do we need to represent? How should we represent them?

Data	How represented
Men, Women	
Preference lists	
Unmatched men	
Who men proposed to	
Engagements	

Jan 19, 2018

Sprenkle - CSCI211

11

Stable Matching Implementation

- What do we need to represent? How should we represent them?

Data	How represented
Men, Women	Integers (ids)
Preference lists	2 Array of arrays (2D array)
Unmatched men	List
Who men proposed to	Integer for each man → Array of integers (maps man's id to "next" woman)
Engagements	2 Arrays, length n

Jan 19, 2018

Sprenkle - CSCI211

12

Asymptotic Analysis of Gale-Shapley Alg

```

Initialize each person to be free
while (some man is free and hasn't proposed to every woman)
    Choose such a man m
    w = 1st woman on m's list to whom m has not yet proposed
    if (w is free)
        assign m and w to be engaged
    else if (w prefers m to her fiancé m')
        assign m and w to be engaged and m' to be free
    else
        w rejects m
  
```

What is the running time of each part of the algorithm?
 What is the total running time of the algorithm?

Jan 19, 2018

Diprimo - CS211

13

Efficient Implementation

- Women rejecting/accepting: determine does woman w prefer man m to man m' ?
 - For each woman, create array of men with her preference
 - inverse* of preference list
 - Constant time access for each query after $O(n)$ preprocessing

Amy	1 st	2 nd	3 rd	4 th	5 th	6 th	7 th	8 th	
Pref	8	3	7	1	4	5	6	2	Contains man's id

For each man, how does he rank?

Amy	1	2	3	4	5	6	7	8	
Inverse	4 th	8 th	2 nd	5 th	6 th	7 th	3 rd	1 st	Amy prefers man 3 to 6 since inverse[3] < inverse[6]

2 7

```

for i = 1 to n
    inverse[ pref[i] ] = i
  
```

Jan 19, 2018

14

Asymptotic Analysis of Gale-Shapley Alg

```

Initialize each person to be free
while (some man is free and hasn't proposed to every woman)
    Choose such a man m
    w = 1st woman on m's list to whom m has not yet proposed
    if (w is free)
        assign m and w to be engaged
    else if (w prefers m to her fiancé m')
        assign m and w to be engaged and m' to be free
    else
        w rejects m
  
```

What is the running time of each part of the algorithm?
 What is the total running time of the algorithm?

Jan 19, 2018

Sprenkle - CSCI211

15

Asymptotic Analysis of Gale-Shapley Alg

Not explicitly in the algorithm, but we need to make the inverse array before the while loop too.

```

Initialize each person to be free     $O(n)$ 
while (some man is free and hasn't proposed to every woman)     $O(n^2)$ 
    Choose such a man m     $O(1)$ 
    w = 1st woman on m's list to whom m has not yet proposed     $O(1)$ 
    if (w is free)     $O(1)$ 
        assign m and w to be engaged     $O(1)$ 
    else if (w prefers m to her fiancé m')     $O(1)$ 
        assign m and w to be engaged and m' to be free     $O(1)$ 
    else
        w rejects m     $O(1)$ 
  
```

Using inverse array

Total: $O(n^2)$

Jan 19, 2018

Sprenkle - CSCI211

16

Looking Ahead

- Wiki due Monday, 11:59 p.m.
 - 2.3
- Problem Set 2 due Friday, before class