COMP4128 Week 01 Workshop

Yifan He

z5173587@unsw.edu.au

Tutorial schedule

Week	Tutor
1-5	Yifan He
7-10	Adam Stucci

About me

- Year 2 PhD student, studying the field of logic & general game playing
  - Advertisement: Prof Michael Thielscher is great
  - If you enjoy SAT/QBF/Logic-related AI, do honours with him
- My competition history (in chronological order)
  - In the 2019 divisional contest (14th place ☹)
  - Did COMP4128 in 19T3, full mark + course prize
  - In the 2020 competition, got regional final 2nd place
  - In the 2021 competition, got regional final 2nd place

Reminder

- Contest 1 (this weekend)
- Problem Set 1

Outline

- What to expect in COMP4128
- Structure of the tutorial sessions
- Problem solving

More information about COMP4128

More information about COMP4128

What COMP4128 is **not** about

- A WAM booster
- An interview course

More information about COMP4128

What COMP4128 is **not** about

- A WAM booster
- An interview course

What you might take away from COMP4128

- Be smarter?
- Smash interview?
- More advanced DS & algorithms
- Practice problem solving in general

About COMP4128 (cont.)

How to do well

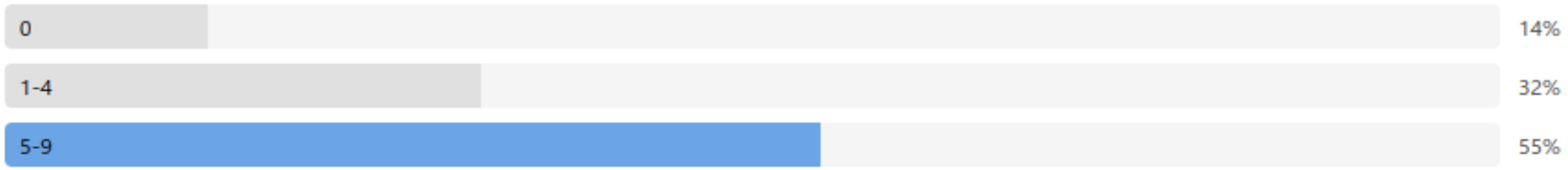
- **Don't** practice leetcode hard problems
- Solve **every** problem set problems **individually**
- Solve similar problems on Codeforces
- You can email me for problem recommendations

Warning

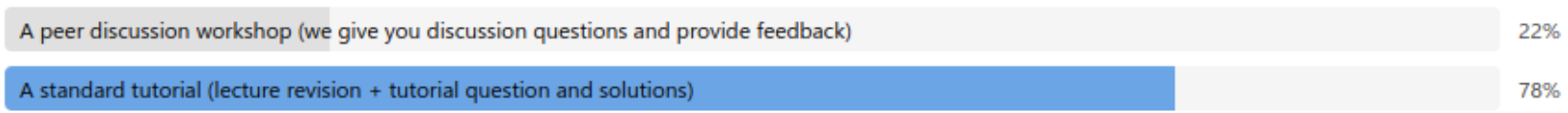
- Don't copy solutions directly
  - Especially, don't spend effort renaming variables and trying to bypass the plagiarism detection system!
- Use contest 1 to "predict" your success rate

Tutorial Structure

1. How many online tutorials (Thursday 12:00-13:30) would you expect to come in the term, there are 9 in total?



2. What's your preferred style of the online tutorial?



- For week 1-5:
  - Will do some revision
  - Solve tutorial question
  - Hints for problem sets
  - A new problem (if we have time)

Adoption (2021 Final exam)

You have N dogs, N cats, and $2N$ people ($N \leq 1e5$). Each person would select exactly one pet. The happiness of person i getting a dog is $d[i]$, and a cat is $c[i]$. Calculate the maximum total happiness of the $2N$ people.

Example (N=2)

c[i]	d[i]
1	2
2	1
0	3
2	-1

c[i]	d[i]
1	2
2	1
0	3
2	5

c[i]	d[i]
1	2
2	1
0	3
-1	-2

Adoption (cont.)

- We can formulate it as an ILP problem
- We introduce $2N$ variables $x[1..2N]$, $x[i] \in \{0,1\}$.
- $x[i] = 1$ (resp. $x[i] = 0$) means person i selects cat (resp. dog).
- Constraint: $\sum_{i=1}^{2N} x[i] = N$.
- Objective: Max $\sum_{i=1}^{2N}(c[i] \cdot x[i] + d[i] \cdot (1 - x[i]))$

Adoption (cont.)

- We can formulate it as an ILP problem
- We introduce $2N$ variables $x[1..2N]$, $x[i] \in \{0,1\}$.
- $x[i] = 1$ (resp. $x[i] = 0$) means person i selects cat (resp. dog).
- Constraint: $\sum_{i=1}^{2N} x[i] = N$.
- Objective: Max $\sum_{i=1}^{2N}(c[i] \cdot x[i] + d[i] \cdot (1 - x[i]))$
- We can simplify the expression to
  - Maximize $\sum_{i=1}^{2N} d[i] + \sum_{i=1}^{2N}(c[i] - d[i]) \cdot x[i]$

Adoption (cont.)

- We can formulate it as an ILP problem
- We introduce $2N$ variables $x[1..2N]$, $x[i] \in \{0,1\}$.
- $x[i] = 1$ (resp. $x[i] = 0$) means person i selects cat (resp. dog).
- Constraint: $\sum_{i=1}^{2N} x[i] = N$.
- Objective: Max $\sum_{i=1}^{2N}(c[i] \cdot x[i] + d[i] \cdot (1 - x[i]))$
- We can simplify the expression to
  - Maximize $\sum_{i=1}^{2N} d[i] + \sum_{i=1}^{2N}(c[i] - d[i]) \cdot x[i]$
- How can we maximize the expression?

Adoption (cont.)

- We can formulate it as an ILP problem
- We introduce $2N$ variables $x[1..2N]$, $x[i] \in \{0,1\}$.
- $x[i] = 1$ (resp. $x[i] = 0$) means person i selects cat (resp. dog).
- Constraint: $\sum_{i=1}^{2N} x[i] = N$.
- Objective: Max $\sum_{i=1}^{2N}(c[i] \cdot x[i] + d[i] \cdot (1 - x[i]))$
- We can simplify the expression to
  - Maximize $\sum_{i=1}^{2N} d[i] + \sum_{i=1}^{2N}(c[i] - d[i]) \cdot x[i]$
- How can we maximize the expression?
- Just select the N largest $c[i] - d[i]$!

Adoption (cont.)

Full solution

- Calculate the sum of $d[1..2N]$ denote it as X
- Calculate $a[i] = c[i] - d[i]$ and sort array a in descending order
- Calculate the sum of $a[1..N]$ denote it as Y
- Answer is $X + Y$

Demo

Complete The Word

Give a string of length N ($N \leq 50,000$), the string is nice if there exists a substring of length 26 where each letter of 'A' to 'Z' appears exactly once. You are given a string where some of its letters are missing (represented with '?'). Is it possible to fill in the missing letters with 'A' to 'Z' so that the resulting string is nice? If yes, find an example.

Example

input: ABC??FGHIJK???OPQR?TUVWXY?
output: ABCDEFGHIJKLMNOPQRSTUVWXYZ

Complete The Word

How to check if a string can be made nice?

Complete The Word

How to check if a string can be made nice?

- There is a substring of length 26 that can be made nice

Complete The Word

How to check if a string can be made nice?

- There is a substring of length 26 that can be made nice
- All letters in the substring are distinct

Complete The Word

How to check if a string can be made nice?

- There is a substring of length 26 that can be made nice
- All letters in the substring are distinct
- Iterate through all substrings of length 26. For each of them check if the letters in it are distinct.

Complete The Word

How to check if a string can be made nice?

- There is a substring of length 26 that can be made nice
- All letters in the substring are distinct
- Iterate through all substrings of length 26. For each of them check if the letters in it are distinct.
- Check can be done with an array. Time complexity O(26*N).
- Can we make it faster?

Complete The Word

How to check if a string can be made nice?

- There is a substring of length 26 that can be made nice
- All letters in the substring are distinct
- Iterate through all substrings of length 26. For each of them check if the letters in it are distinct.
- Check can be done with an array. Time complexity O(26*N).
- Can we make it faster?
- Yes, sliding windows!

Complete The Word

How to check if a string can be made nice?

- There is a substring of length 26 that can be made nice
- All letters in the substring are distinct
- Iterate through all substrings of length 26. For each of them check if the letters in it are distinct.
- Check can be done with an array. Time complexity O(26*N).
- Can we make it faster?
- Yes, sliding windows!
- Maintain a window of size 26, and dynamically maintain (#distinct letters, #letters), and check they are equal. Time complexity O(N).

Complete The Word

How to complete the string to be nice?

Complete The Word

How to complete the string to be nice?

- Locate the "all distinct" substring

Complete The Word

How to complete the string to be nice?

- Locate the "all distinct" substring
- Detect the missing letters

Complete The Word

How to complete the string to be nice?

- Locate the "all distinct" substring
- Detect the missing letters
- Replace the ’?’ with the missing letters

Complete The Word

How to complete the string to be nice?

- Locate the "all distinct" substring
- Detect the missing letters
- Replace the ’?’ with the missing letters
- Don’t forget about ’?’ in other substrings.

Complete The Word

How to complete the string to be nice?

- Locate the "all distinct" substring
- Detect the missing letters
- Replace the ’?’ with the missing letters
- Don’t forget about ’?’ in other substrings.
- Just fill it with ’A’.

Demo

Match Points [1]

You are given a set of points $x_1, x_2, ..., x_n$ on the number line. Two points i and j can be matched with each other if the following conditions hold:

- neither i nor j is matched with any other point;
- $|x_i - x_j| \geq z$.

What is the maximum number of pairs of points you can match with each other?

- $2 \leq n \leq 200,000$ and $1 \leq z \leq 10^9$

[1] https://codeforces.com/contest/1156/problem/C

Match Points [1]

You are given a set of points $x_1, x_2, ..., x_n$ on the number line. Two points i and j can be matched with each other if the following conditions hold:

- neither i nor j is matched with any other point;
- $|x_i - x_j| \geq z$.

What is the maximum number of pairs of points you can match with each other?

- $2 \leq n \leq 200,000$ and $1 \leq z \leq 10^9$

Example

n = 4, z = 2

x = [1, 3, 3, 7]

[1] https://codeforces.com/contest/1156/problem/C

Match Points

Can we do greedy matching?

- Obvious first step is sorting

Match Points

Can we do greedy matching?

- Obvious first step is sorting
- Iterate from the largest x_n, match it with the largest x_i such that $|x_n - x_i| \geq z$?

Match Points

Can we do greedy matching?

- Obvious first step is sorting
- Iterate from the largest x_n, match it with the largest x_i such that $|x_n - x_i| \geq z$?
- Wrong answer!
- N = 6, Z = 9
- 1 2 3 10 13 19
- But a better solution is 1 2 3 10 13 19

Match Points

A Modified Problem

Match Points

A Modified Problem

- Can you check if there exists a match of size K?

Match Points

A Modified Problem

- Can you check if there exists a match of size K?
- $K \leq N/2$

Match Points

A Modified Problem

- Can you check if there exists a match of size K?
- $K \leq N/2$
- There is a match of size K iff the match $(x_1, x_{N-K+1}), (x_2, x_{N-K+2}), ..., (x_K, x_N)$ is good.

Match Points

A Modified Problem

- Can you check if there exists a match of size K?
- $K \leq N/2$
- There is a match of size K iff the match $(x_1, x_{N-K+1}), (x_2, x_{N-K+2}), ..., (x_K, x_N)$ is good.
- Can be proved by the exchange argument

Match Points

A Modified Problem

- Can you check if there exists a match of size K?
- $K \leq N/2$
- There is a match of size K iff the match $(x_1, x_{N-K+1}), (x_2, x_{N-K+2}), ..., (x_K, x_N)$ is good.
- Can be proved by the exchange argument
- If there is a match of size K, can we find a match of size K-1?

Match Points

A Modified Problem

- Can you check if there exists a match of size K?
- $K \leq N/2$
- There is a match of size K iff the match $(x_1, x_{N-K+1}), (x_2, x_{N-K+2}), ..., (x_K, x_N)$ is good.
- Can be proved by the exchange argument
- If there is a match of size K, can we find a match of size K-1?
- Binary search to find the maximum K!

Demo