arcanae's blog

How to determine the solution of a problem by looking at its constraints?

By arcanae, 6 years ago,

By looking at the constraints of a problem, we can often "guess" the solution.

Common time complexities

Let n be the main variable in the problem.

- If $n \le 12$, the time complexity can be O(n!).
- If $n \le 25$, the time complexity can be $O(2^n)$.
- If $n \le 100$, the time complexity can be $O(n^4)$.
- If $n \le 500$, the time complexity can be $O(n^3)$.
- If $n \le 10^4$, the time complexity can be $O(n^2)$.
- If $n \le 10^6$, the time complexity can be O(n log n).
- If $n \le 10^8$, the time complexity can be O(n).
- If $n > 10^8$, the time complexity can be O(log n) or O(1).

Examples of each common time complexity

- O(n!) [Factorial time]: Permutations of 1 ... n
- $O(2^n)$ [Exponential time]: Exhaust all subsets of an array of size n
- $O(n^3)$ [Cubic time]: Exhaust all triangles with side length less than n
- $O(n^2)$ [Quadratic time]: Slow comparison-based sorting (eg. Bubble Sort, Insertion Sort, Selection Sort)
- O(n log n) [Linearithmic time]: Fast comparison-based sorting (eg. Merge Sort)
- O(n) [Linear time]: Linear Search (Finding maximum/minimum element in a 1D array), Counting Sort
- O(log n) [Logarithmic time]: Binary Search, finding GCD (Greatest Common Divisor) using Euclidean Algorithm
- O(1) [Constant time]: Calculation (eg. Solving linear equations in one unknown)

Explanations based on Codeforces problems

1. 255D Mr. Bender and Square
   Observe that $1 \le n, c \le 10^9$. Referring to the information above, the program's time complexity should be either O(log n) or O(1). Since no O(1) solution exists, we conclude that binary search must be used.
2. 580B Kefa and Company
   In this problem, $1 \le n \le 10^5$, which suggests that the time complexity can be either O(n log n) or O(n). It is quite obvious that sorting is required. Therefore, O(n log n) is the correct solution of this problem.
3. 583B Robot's Task
   Notice that n in very small ($1 \le n \le 1000$) in this problem. It means that a $O(n^2)$ solution can solve it. We simply need to simulate the robot's moves.

I hope that this can help you figure out the solution of some problems quicker :)

Note: The above method may not always work in all problems. Some may require algorithms that have complex time complexities, while in some problems like 591B Rebranding, the range of n does not match the time complexity of the "optimal" solution. ($1 \le n, m \le 200\,000$ suggests that the time complexity is O(n log n) or O(n) but the time complexity of the solution is actually O(1).)

time complexity, solution, constraint

+46 arcanae 6 years ago 12

Comments (12)

Write comment?

Zlobober 6 years ago, # | +17

10^8 for $O(n \log n)$ sounds too optimistic. I'd say, for $n \le 10^6 O(n \log n)$ has a chance to pass.

→ Reply

Hohol 6 years ago, # | +8

Hey, why do you call mergesort non-comparison-based?

→ Reply

Xellos 6 years ago, # | +5

Why do you care about lower bounds? Only upper bounds matter (except maybe some special cases / undefined-ness).

If it's $n \le 50$, it can be anything polynomial, but there's probably a simple slower solution and a faster one that's harder to implement. Also, you have 75 minutes.

→ Reply

arcanae 6 years ago, # ^ | +3

What do you mean by "75 minutes"?

→ Reply

Xellos 6 years ago, # ^ | +3

Old TC, of course.

→ Reply

arcanae 6 years ago, # | 0

Thanks for pointing out my mistakes. I have changed them.

→ Reply

wrick 6 years ago, # | ← Rev. 2 0

Another good way to remember this is that whatever algorithm you chose, if you plug in `n` , you should get around `~10^8` . This makes is easier to analyze where your complexity might depend on 2 things e.g. `O(MN log N)`

```
log10 (n!)      = 8.68 for n = 12
log10(2^n)      = 7.52 for n = 25
log10(n^4)      = 8.00 for n = 100
log10(n^3)      = 8.09 for n = 500
log10(n^2)      = 8.00 for n = 10^4
log10(n log n)  = 7.29 for n = 10^6
log10(n)        = 8.00 for n = 10^8
```

→ Reply

cooper71 18 months ago, # | 0

some problems have multiple test cases.As like 1<=t<=100000 and 1<=n<=1000000. At this type of problem how I can understand what algorithm will be needed?

→ Reply

calebeof 18 months ago, # ^ | +5

These problems often say that the sum of all cases is, at most, *N*.

→ Reply

kevalshah90909 18 months ago, # | 0

Should u consider value of each element in the constarint ?

→ Reply

Qg3 3 months ago, # | 0

When `N <= 10^5`

We can use `O(NVN)` some kind of brute force or any square root algorithm like Mo's.

Problem for this senario

→ Reply

→ Pay attention

Before contest
Codeforces Round #750 (Div. 2)
2 days

→ hotsonhonet

Rating: 898
Contribution: 0

- Settings
- Blog
- Teams
- Submissions
- Talks
- Contests

hotsonhonet

→ Top rated

#	User	Rating
1	tourist	3707
2	Benq	3672
3	Radewoosh	3627
4	ksun48	3547
5	Miracle03	3480
6	ecnerwala	3400
7	peehs_moorhsum	3384
8	maroonrk	3361
9	sunset	3338
10	Um_nik	3320

Countries | Cities | Organizations View all →

→ Top contributors

#	User	Contrib.
1	1-gon	209
2	Um_nik	196
3	YouKn0wWho	192
4	Errichto	182
5	sus	180
5	awoo	180
7	tourist	175
8	-is-this-fft-	171
9	SecondThread	170
9	Radewoosh	170

View all →

→ Find user

Handle:
Find

→ Recent actions

MikeMirzayanov → PBOX — simple package manager for Windows (written and used in Codeforces)
AIFlen → Codeforces Round #750 (Div. 2)
smax → [Tutorial] Berlekamp-Massey
maverick_GOD → Invitation for CODE QUEST 2021
MiptLited → Moscow International Workshop 2021 starts soon!
v0rtex → Minimum subset size for target sum.
SD_Olers → I would like to ask a question
CodeChef_admin → SnackDown 2021 is back! Registrations are now open
rangeupdate → Codeforces Problem Used in Provincial Contest
triple__a → Codeforces Round #630 Editorial
Zhora_.004 → TLE in C++ 20
Eric12 → Hi,Codeblocks
stefdasca → Offering high quality and affordable group classes
underdog. → Google Interview Question
jonathanirvings → Do touch everything
fie_claussell → Is it possible to update test data of past USACO contest and where to report it?
codemastercpp → Yet another youtuber
Tlatoani → Editorial for Technocup 2022 — Elimination Round 1 and Codeforces Round #749 (Div. 1+Div. 2)
Utkarsh.25dec → Invitation to CodeChef SnackDown 2021 Round 1A
MikeMirzayanov → C++20 Is Released
MohamedAboOkail → Codeforces Extensions
ocelot → How to solve this problem (K burners and N gas cylinders)
CodeChef_admin → Last few hours of CodeChef Snackdown Qualifiers!
Hernan → Far Manager as IDE
-Omar_Hafez- → [Competitive Programming] A new tool for competitive programmers

Detailed →