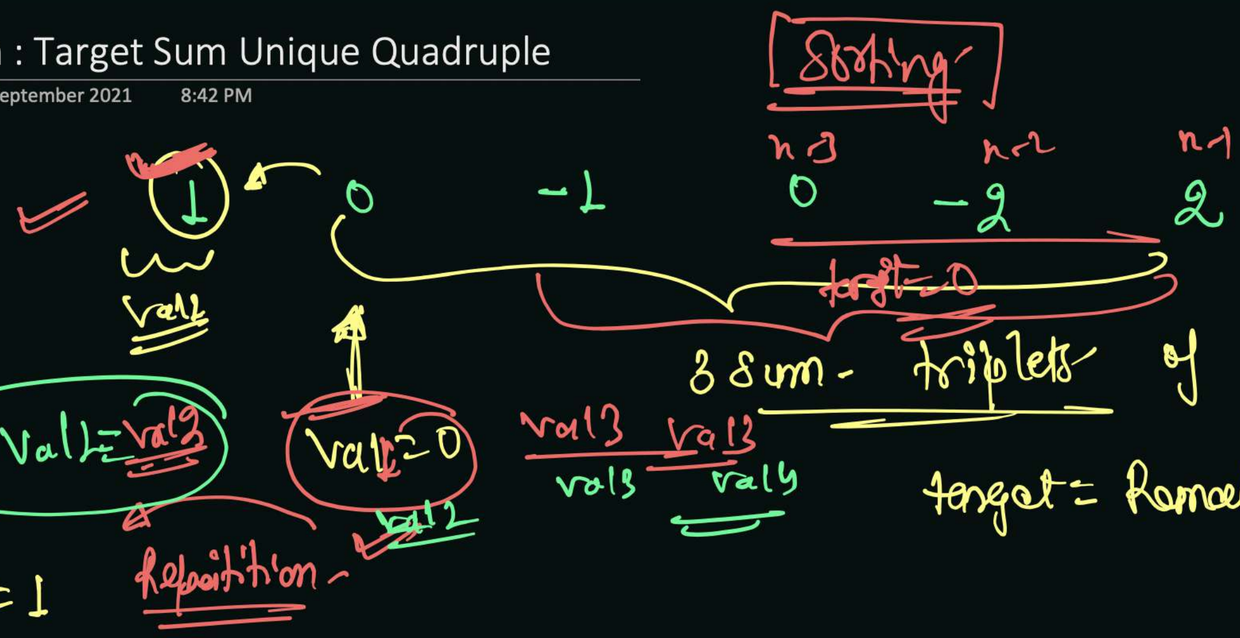
4 sum : Target Sum Unique Quadruple

Sunday, 5 September 2021 8:42 PM



target = 0

Sorting

→ Prevent form Repitition

Quadruple → Hoo → check of previous Element

val1 + val2 + val3 + val4 = target

target = Remaing target

val1 = 1

Remaining target = target - val1

= 0 - (1)

= -1

triplet → $\begin{bmatrix} 0, & -1, & 0, & 1 \\ -1, & -2, & 2, & 1 \end{bmatrix}$ with val1 = 1

val1 = 0

Remaing target = target - val1

= 0 - 0

= 0

$\begin{bmatrix} 0 & -2, & 2, & 0 \end{bmatrix}$ with val1 = 0

→ ① Sorting -

→ Algo

② val1 Considered in in loop from i=0 to <n-3

K sum : Target Sum Unique Set

Sunday, 5 September 2021 8:42 PM

Time → $(n^{k-1} + n\log n)$ if $k = 2$

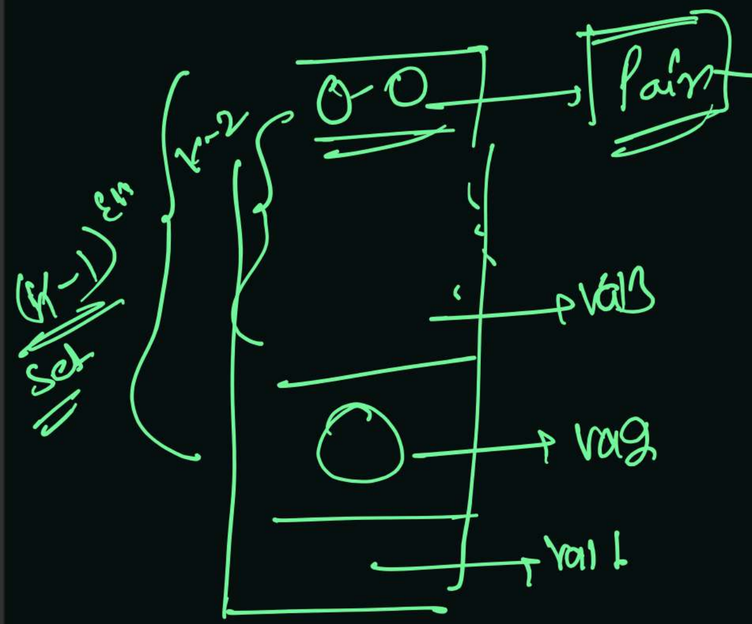
K values

sum = target

→ array.

val1

k-1 vals sum = target - val1



Base problem → Base case of Recursion of k sum

Set 1 + val1

Set 2 + val1

Set 3 + val1

Set 4 + val1

Record

if(k=2) Solve pair problem

for(int i = si; i < n-k; i++){

k-1 → set

}

Sieve of Eratosthenes

Thursday, 9 September 2021 9:22 PM

complexity to find prime Number → $O(\sqrt{n})$

upperRange → Query → 0 to upperRange-

of is prime-

(q)

Complexity → $q \times \sqrt{n}$

→ val = upperRange

q time

$q \times \sqrt{n}$

$q >> n$

↳ Reduce complexity in → $O(q)$

Note: Sieve of eratosthenes can find is prime in $O(1)$ time complexity.

How ??

n=30 — find all prime between 0 to 30

we sacrifice with space to reduce time complexity.

make a boolean array of size = n+1

0 → F

1 → F

2 → T

3 → T

4 → F

5 → T

6 → F

7 → T

8 → F

9 → F

10 → F

11 → T

12 → F

13 → T

14 → F

15 → F

16 → F

17 → T Prime

18 → F

19 → T

20 → F

21 → F

22 → F

23 → T

24 → F

25 → F

26 → F

27 → F

28 → F Not prime

29 → T

30 → F

4 is not prime because it is multiple of 2

All factors of 4 is already marked

arr[i] == True. ⇒ i is prime no or not prime

To check if 30 is prime or not,

Factor

2

3

4 → X Not prime

5

√n

Prime

9 will mark it's factor

2 to √n

u – upper range

Complexity → Preprocessing time complexity → $O(n \log\log n)$ → complexity of query ↓ $O(1)$

2 3 4 5

$= \frac{n}{2} + \frac{n}{3} + \frac{n}{5} + \cdots + \frac{n}{\sqrt{K}}$ → max prime no. b/w 2 to $\sqrt{n}$

$= n\left[\frac{1}{2} + \frac{1}{3} + \frac{1}{5} + \cdots\right]$

2 to $\sqrt{30}$ → 2 3 5 ; $n = 30$

$\frac{3x}{7} = 5$

$\sum \frac{n}{i}$ where 'i' is prime b/w 2 to $\sqrt{K}$ = Sum of inverse of prime number

→ Convergence - divergence → concept

$= n\left[\log(\log(n))\right]$ /// nearly 1 [at most = 5]

So overall time complexity → $n\left(\log\log n\right) \equiv O(n)$

$= 5n \equiv n$

$\log\log$

$10^9 \equiv 2^{32} \equiv 5$

$\log\log(10^9) =$

$\log\log(2^{32})$

$\log(32) = \log(2^5)$

$5 \times 10^1 =$ 5 10

Steps to code Sieve Algorithm.

① Make a boolean array of size n+1: → By default → False.

② Mark all number as prime i.e. arr[i] = true.

③ Travel from 2 to $\sqrt{n}$, if arr[i] == True.

then mark its multiple as not prime. [→ arr[factor] = false]

if (arr[i] == false] → continue.

if it is marked (then), its factor are already marked.

④ Return is prime array.

False → Prime

True is used as marker

Segmented Sieve

Thursday, 9 September 2021 10:41 PM

allowed Time complexity ⇒ $(b-a)\log\log(b-a)$

Approach - 0 → Check Every Number in query if it is prime or not. ✗

↳ May be it will pass test cases but not match with Required time compl.

Time complexity → $q \times \sqrt{b}$

Approch - 1 → Sieve → make of size → $b+1$

↳ constraints.

① $1 \le a \le b \le 10^9$ → Range of a & b

↳ we can't make size of array of b. because max. allowed size in array is of order 10^5.

② $b - a \le 10^5$

→ Not even pass ~~test~~ case

Missing Constraints

Segmented Sieve →

Lower Range = a Upper Range = b.

$b-a \leq 10^5$

$1 \leq a \leq b \leq 10^9$

a = 22 , b = 51

Size of = $b-a+1 = 51-22+1 = 30$

a and b both are included.

(2) → prime no. b/w a to $\sqrt{b}$. ($\sqrt{b} \rightarrow 10^5$) (using sieve)

arr[i] == False. → i is prime no.

(3) → marking of not prime
↳ begin with starting index of multiple of prime no.

a to $\sqrt{51}$ → 2, 3, 5, 7

Base value = a

Index | Val

0 → 22 F T

1 → 23 F

2 → 24 F T T

3 → 25 F T

4 → 26 F T

5 → 27 F T

6 → 28 F T T

7 → 29 F

8 → 30 F T T T

9 → 31 F

10 → 32 F T

11 → 33 F T

12 → 34 F T

13 → 35 F T T

14 → 36 F T T

15 → 37 F

16 → 38 F T

17 → 39 F T

18 → 40 F T T

19 → 41 F

20 → 42 F T T

21 → 43 F

22 → 44 F T

23 → 45 F T T

24 → 46 F T

25 → 47 F

26 → 48 F T

27 → 49 F T

28 → 50 F T

29 → 51 F T

Val = 0 index + Base value : 17 + 22 = 39

How to find Starting Index of multiple of prime Number.

prime → 2, 3, 5, 7

multiple = $\left[\frac{q * 1.0}{prime[i]}\right]_{ceil}$

$m = \left[\frac{22 \times 1.0}{7}\right]_{ceil} = [3.1\text{--}]_{ceil} = 4$

Imp: multiple = 1] multiple++

Starting Index = multiple * prim[i] − a3 [i−

$8i = 4 * 7 - 22$

28

$= c$

Exception: → 2 (marks 4 is not prime)

prime → 2, 3, 5, ...

b.

multiple = $\left[\frac{2 * 1.0}{2}\right]_{ceil} = 1$

8i Index = 1 * 2 − 2

8i = 0

Steps to Solve Segmented sieve-

$\int_0^5 = 10^4$ → Sieve

(1) Get prime no b/w 2 to $\sqrt{b}$ using sieve

(2) Solve b/w b-a array, and find starting index as:

$$multiple = \left\lceil \frac{a}{prime[i]} \right\rceil_{ceil}$$ if multiple == 1 multiple ++,

$s.i = multiple * prime[i] - a$

(3) get value of prime using.

$val = index + \underbrace{Base\ value}_{a} = index + a$

Pair with difference K

Friday, 10 September 2021 12:04 AM

5 20 3 2 5 80 , diff = 78 = k

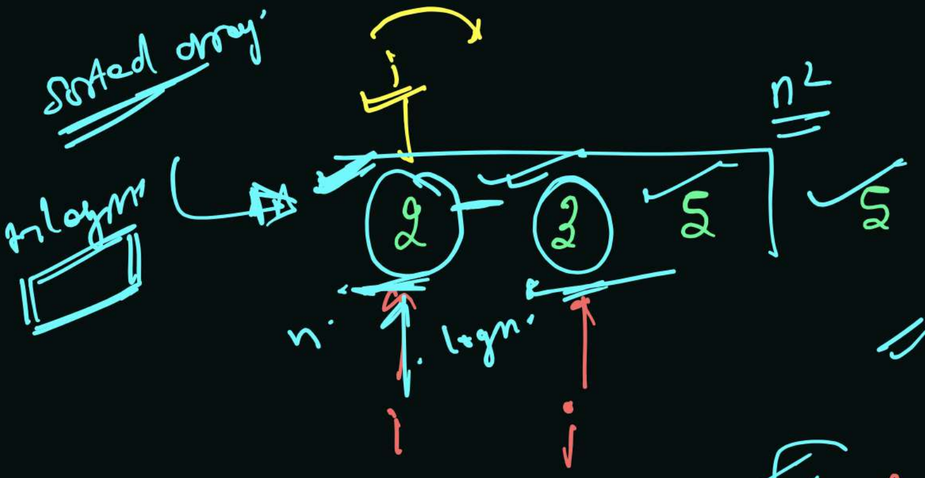
Sum.

① Sort the array.

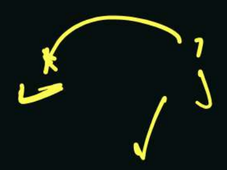
② Make two pointer i=0, j=1;

③ Solve.

Sorted array:



80



298

20 80 800

diff = arr[j] - arr[i]

diff = arr[


```
if (diff == k) {
    return true;
} else if( diff > k) { Reduce diff
    i++;
} else { // diff < k  Increase diff.
    j++;
}
```