



Zadanie 1

Dane są dwa zbiory liczb, reprezentowane jako tablice rozmiarów m i n , gdzie m jest znacznie mniejsze od n . Zaproponuj algorytm, który sprawdzi, czy zbiory są rozłączne.

T_m
 $m < n$
 T_n

<div>1</div>	<div>4</div>	<div>7</div>	<div>3</div>	
5	3	8	13	15

$O(mn)$

$T_m \rightarrow \text{sort} \rightarrow O(m \log m)$

1	3	4	7
	/ False		

$T_n \xrightarrow{\text{binsearch}}$

5	3
---	---

✓ 1) sort $\rightarrow O(m \log m)$
2) binsearch
Use very? $\rightarrow O(n)$
Is it? $\rightarrow O(\log m)$
 $O(n \log m)$
 $\rightarrow O((n+m) \log m)$

$T_m \rightarrow \text{sort} \rightarrow 1$

3	4	7	
5	8	13	15

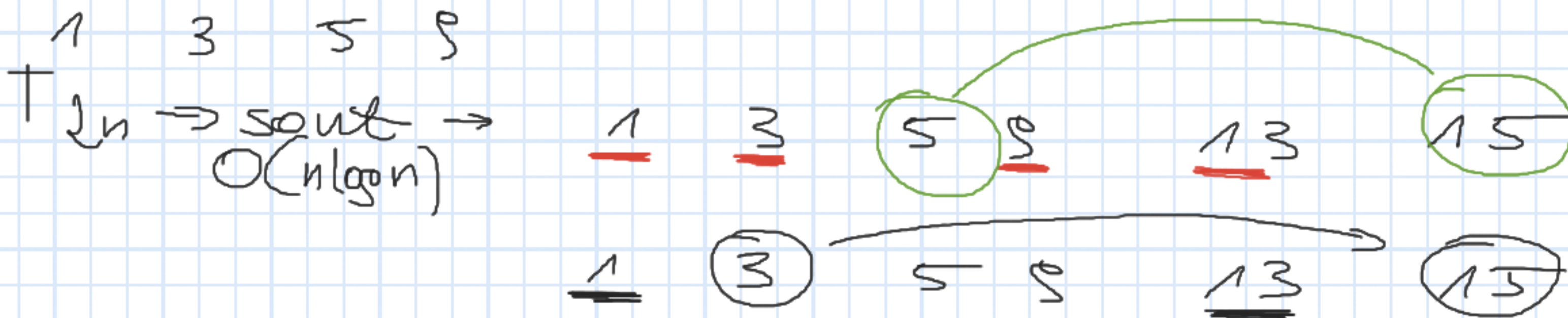
$T_n \rightarrow \text{sort} \rightarrow 2$

$n \log n + m \log m + \cancel{n} + \cancel{m} = O(n \log n + m \log m)$



Zadanie 2

Dana jest tablica $2n$ liczb rzeczywistych. Zaproponuj algorytm, który podzieli te liczby na n par w taki sposób, że podział będzie miał najmniejszą maksymalną sumę liczb w parze. Przykładowo, dla liczb $(1, 3, 5, 9)$ możemy mieć podziały $((1,3),(5,9))$, $((1,5),(3,9))$, oraz $((1,9),(3,5))$. Sumy par dla tych podziałów to $(4, 14)$, $(6, 12)$ oraz $(10, 8)$, w związku z tym maksymalne sumy to 14 , 12 oraz 10 . Wynika z tego, że ostatni podział ma najmniejszą maksymalną sumę.





Zadanie 5

Dana jest posortowana rosnąco tablica A wielkości n zawierająca parami różne liczby naturalne. Podaj algorytm, który sprawdzi, czy jest taki indeks i , że $A[i] == i$.

Co zmieni się, jeżeli liczby będą po prostu całkowite, niekoniecznie naturalne?

$\# f$
 $[\dots]$ $O(\log n)$

$[-4, -2, -1, 3, 5, 9]$
↑ ↑ ↑
 i v i v
 $2 > -1$ $4 < 5$



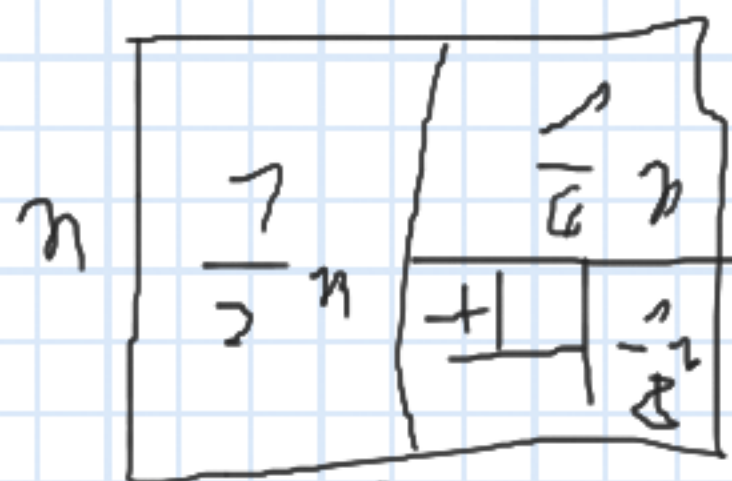
Zadanie 6

Dane jest n punktów na osi liczbowej jednowymiarowej. Napisz algorytm, który stwierdzi, w którym z nich należy wybudować dom, tak aby suma euklidesowych odległości od tego punktu do wszystkich pozostałych była minimalna. Należy zwrócić również tę sumę. Algorytm powinien być jak najszybszy.

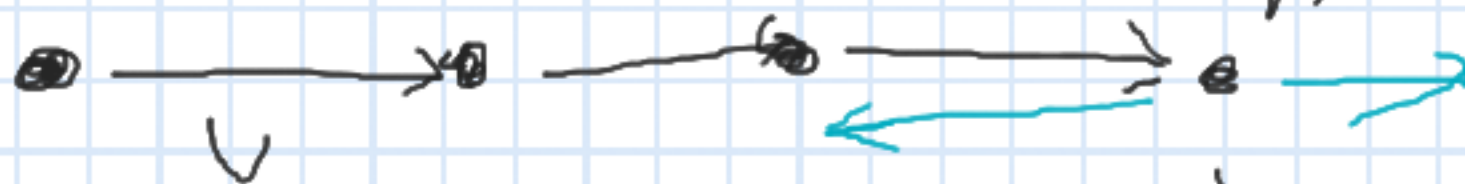
$[1, 5, 7, 2, 3, 8, 6]$

↑

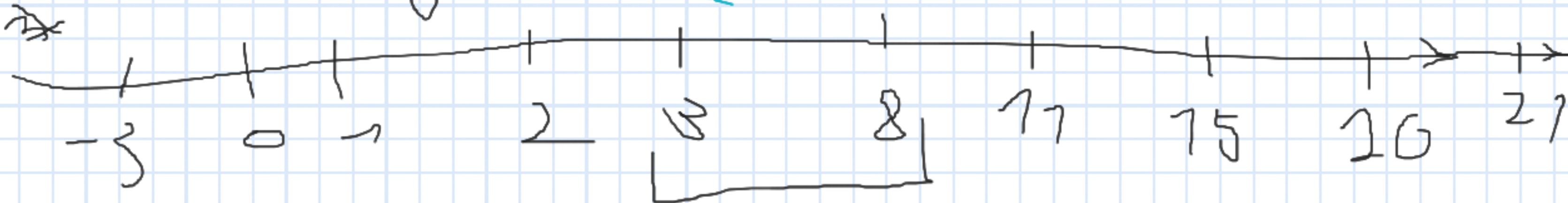
$[1, 2, \dots, \dots]$



$$+2v - \cancel{7v} = -5v$$



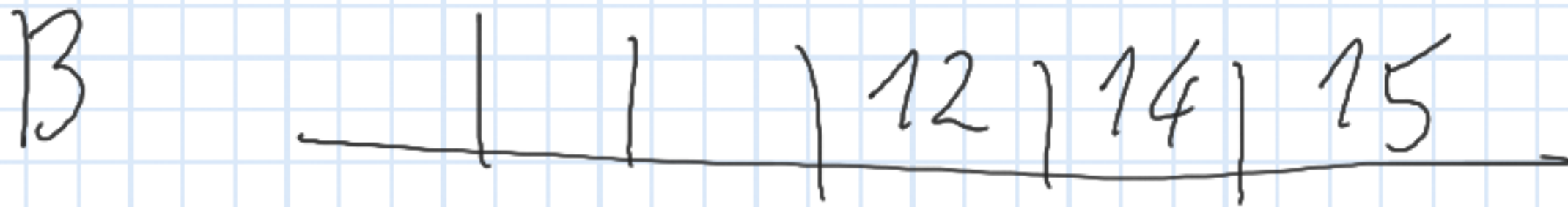
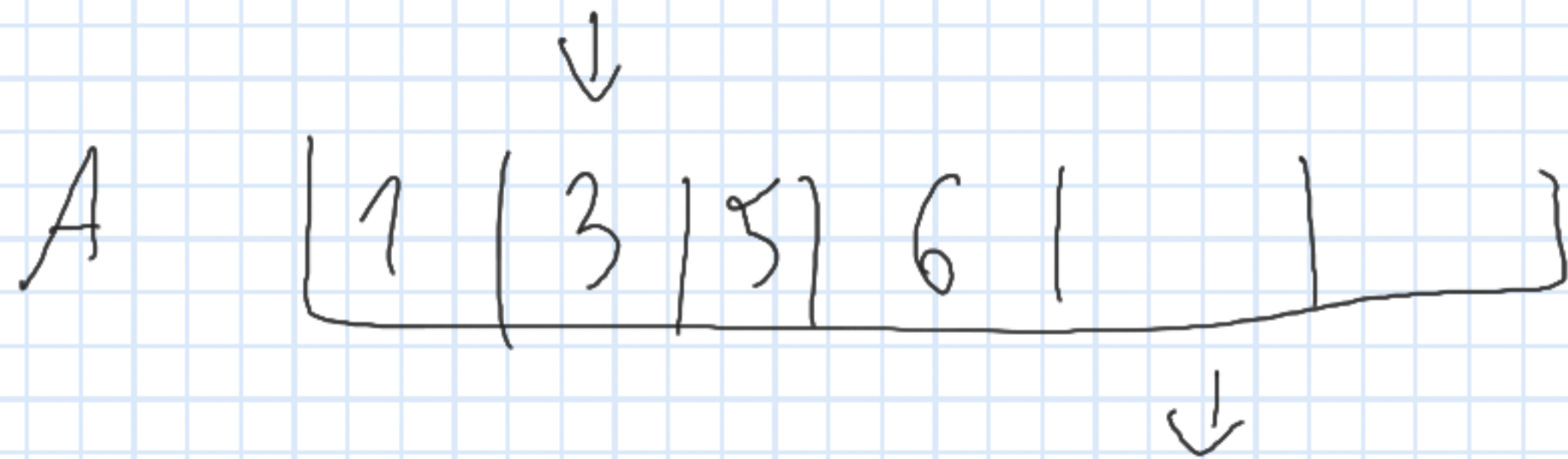
$$n = 2k + 1 \quad O(n)$$
$$n = 2k \quad 4n = O(n)$$



Zadanie 7

Dane są trzy zbiory reprezentowane przez tablice: A, B i C. Napisz algorytm, który powie, czy istnieje taka trójka a, b, c z odpowiednio A, B, i C, że $a + b = c$. Nie wolno korzystać ze słowników!

1. Posortować A i B
2. $\forall c$



$$c = 15$$

$$1 + 15 = 16 > c$$

$$1 + 14 = 15 = c$$

$$c = 17$$

$$16 < c$$

$$3 + 15 = 18 > c$$

$$3 + 14 = 17 = c$$

$$O(a \log a + b \log b + c \cdot (a + b))$$

Zadanie 8

Dana jest tablica A oraz liczba k. Znaleźć liczbę różnych par elementów z tablicy A o różnicy równej k.

$$a \rightarrow a+k$$

Przykład: Dla tablicy [7,11,3,7,3,9,5] oraz k = 4 odpowiedź to 3

$$(7, 11) \quad (3, 7) \quad (9, 5)$$

$$\downarrow \\ [3, 5, 7, 9, 11]$$

1. Posortować i usunąć duplikaty

2. $\forall a$ find $a+k$ with bin Search $a = 3$ $a+k = 7$

$$O(n \log n + n + n \cdot \log n) = O(n \log n)$$

Zadanie 9 (kolokwium)

Cyfra jednokrotna to taka, która występuje w danej liczbie dokładnie jeden raz. Cyfra wielokrotna to taka, która w liczbie występuje więcej niż jeden raz.

Mówimy, że liczba naturalna A jest ładniejsza od liczby naturalnej B, jeżeli w liczbie A występuje więcej cyfr jednokrotnych niż w B, a jeżeli cyfr jednokrotnych jest tyle samo to ładniejsza jest ta liczba, która posiada mniej cyfr wielokrotnych. Na przykład: liczba 123 jest ładniejsza od 455, liczba 1266 jest ładniejsza od 114577, a liczby 2344 i 67333 są jednakowo ładne.

Dana jest tablica T zawierająca liczby naturalne. Proszę zaimplementować funkcję: `pretty_sort(T)`, która sortuje elementy tablicy T od najładniejszych do najmniej ładnych. Użyty algorytm powinien być możliwie jak najszybszy. Proszę w rozwiązaniu umieścić 1-2 zdaniowy opis algorytmu oraz proszę oszacować jego złożoność czasową.

1 2 3

4 5 5

S: 3

1

m: 0

2

1 2 6 6

1 1 4 5 7 7

S: 2

2

m: 1

2

(S, m)

$O(n)$

