

Today's Content

1. Printing Next Permutation
2. Wave array.

Permutations: #Diff arrangement

1: c e a

2: c a e

3: e a c

4: e c a

5: a e c

6: a c e

Write permutations
in lexicographical

1: a c e

2: a e c

3: c a e

4: c e a

5: e a c

6: e c a

Q Given an arrangement, next permutation in lexicographical order

Dictionary Order

Given 10 14 5 : Next Permutation lex: 14 5 10

Given 14 5 10 : Next Permutation lex: 14 10 5

Given 14 10 5 : Next Permutation lex: -1;

Ex:

All permutations of 10 14 5

5 10 14

5 14 10

10 5 14

10 14 5

14 5 10

14 10 5

Given an array find the next permutation of array in Dictionary Order

Ex1: $arr[] = \{1\ 2\ 3\ 4\}$ #output = $\{1\ 2\ 4\ 3\}$

Ex2: $arr[] = \{2\ 1\ 3\}$ #output = $\{2\ 3\ 1\}$

Ex3: $arr[] = \{4\ 3\ 2\}$ #output = $\{-1\}$

#obs: If Data is decreasing order, no next permutation

Ex4: $arr[] = \{8\ 10\ 7\ 6\ 5\} \rightarrow \{10\ 8\ 7\ 6\ 5\} \rightarrow \{10\ 5\ 6\ 7\ 8\}$
↑ reverse it

#obs1: After 8, data decreasing, means with 8, we cannot start any other permutations

#obs2: Swap 8 with smallest element greater than 8

Ex5: $arr[] = \{7\ 12\ 9\ 4\ 2\ -1\} \rightarrow \{9\ 12\ 7\ 4\ 2\ -1\} \rightarrow \{9\ -1\ 2\ 4\ 7\ 9\}$
reverse it

Ex6: $arr[] = \{9\ 18\ 15\ 12\ 10\ 6\ 4\ 2\ 0\}$

reverse it
 $\{10\ 18\ 15\ 12\ 9\ 6\ 4\ 2\ 0\}$

$\{10\ 0\ 2\ 4\ 6\ 9\ 12\ 15\ 18\}$

Ex7: $arr[] = \{3\ 5\ 11\ 7\ 20\ 18\ 14\ 10\ 6\ 3\ 1\}$
0 1 2 3 4 5 6 7 8 9 10
x x x ✓

3 5 11 10 20 18 14 7 6 3 1
x x x ✓ reverse it

3 5 11 10 1 3 6 7 14 18 20
x x x ✓ reverse it

Ex: 0 1 2 3 4 5 6 7 8 9 10
 $arr[] = \{19, 14, 7, 6, 20, 18, 13, 12, 10, 4, 2\}$

#Step1: Find index i , such that $[i+1..N-1]$ data is decreasing?

int $i = N-2;$

while($arr[i] > arr[i+1]$) {

$i--;$

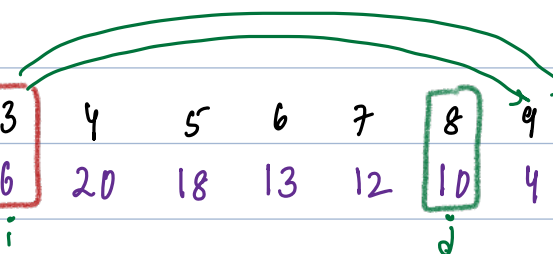
if($i == -1$) { # No next permutation return -1;

return arr;

return arr;

}

#arr: {8 6 4 3}

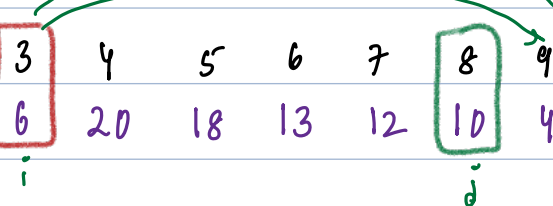
0 1 2 3 4 5 6 7 8 9 10
 $arr[] = \{19, 14, 7, 6, 20, 18, 13, 12, 10, 4, 2\}$


#Step2: Find ceil of $arr[i]$, in $[i+1..N-1]$

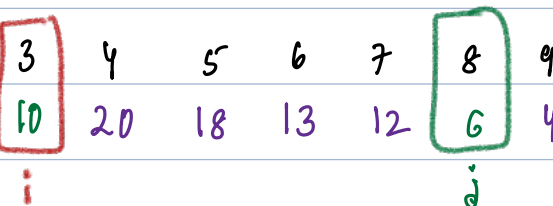
int $j = N-1;$

while($arr[i] > arr[j]$) {

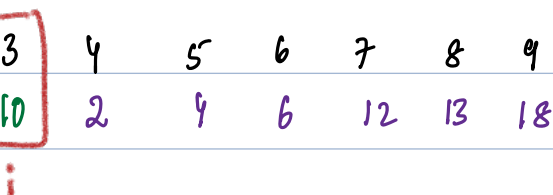
$j--;$

0 1 2 3 4 5 6 7 8 9 10
 $arr[] = \{19, 14, 7, 6, 20, 18, 13, 12, 10, 4, 2\}$


#Step3: #Swap $arr[i]$ & $arr[j]$;

0 1 2 3 4 5 6 7 8 9 10
 $arr[] = \{19, 14, 7, 10, 20, 18, 13, 12, 6, 4, 2\}$


#Step4: #Reverse $arr[]$ from $i+1..N-1$;

0 1 2 3 4 5 6 7 8 9 10
 $arr[] = \{19, 14, 7, 10, 2, 4, 6, 12, 13, 18, 20\}$


vector<int> Permutation(vector<int> &arr) { T.C: $O(3^{N+1}) = O(3^N)$

28 Wave Array

Given $arr[N]$ elements re-arrange array in a wave form such that $arr[0] \geq arr[1] \leq arr[2] \geq arr[3] \leq arr[4] \geq arr[5] \geq \dots$

Note: If multiple right answers are possible return any one of them.

Ex1: $arr[] = \{5 \quad 10 \quad 12 \quad 14 \quad 20 \quad 24\}$

Output1 = $\{10 \quad 5 \quad 14 \quad 12 \quad 24 \quad 20\}$ #Note: All of them are valid.

Output2 = $\{12 \quad 5 \quad 14 \quad 10 \quad 24 \quad 20\}$

Output3 = $\{12 \quad 5 \quad 24 \quad 10 \quad 20 \quad 14\}$

Ex2: $arr[] = \{7 \quad 9 \quad 10 \quad 14 \quad 9 \quad 8 \quad 7\}$

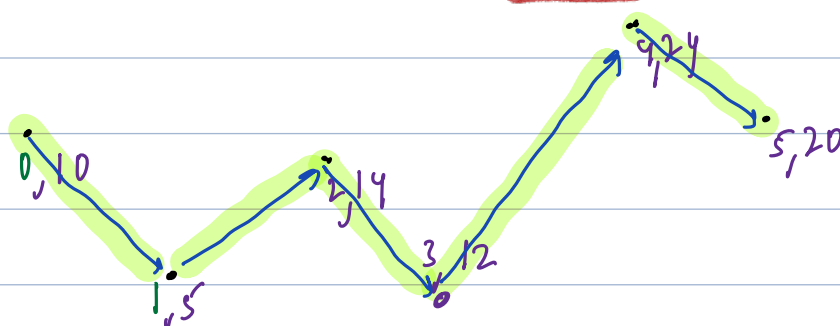
Output = $\{9 \quad 7 \quad 10 \quad 8 \quad 14 \quad 7 \quad 9\}$

#Ideal 1. Sort $arr[]$

2. Swap pair of elements $arr[i]$ & $arr[i+1]$ & update $i = i+2$

$arr[] = \{5 \quad 12 \quad 20 \quad 14 \quad 10 \quad 24\}$

$arr[] = \{ \boxed{5 \quad 10} \quad \boxed{12 \quad 14} \quad \boxed{20 \quad 24} \}$



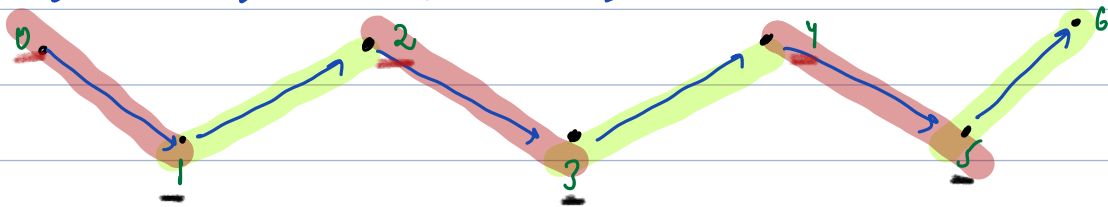
TC: $O(N \log N)$ SC: $O(1)$

#obs: Sorting approach will give us lexicographically 1st wave array.

#Idea2:

Expected arrangement:

$a[0] > a[1] < a[2] > a[3] < a[4] \dots$

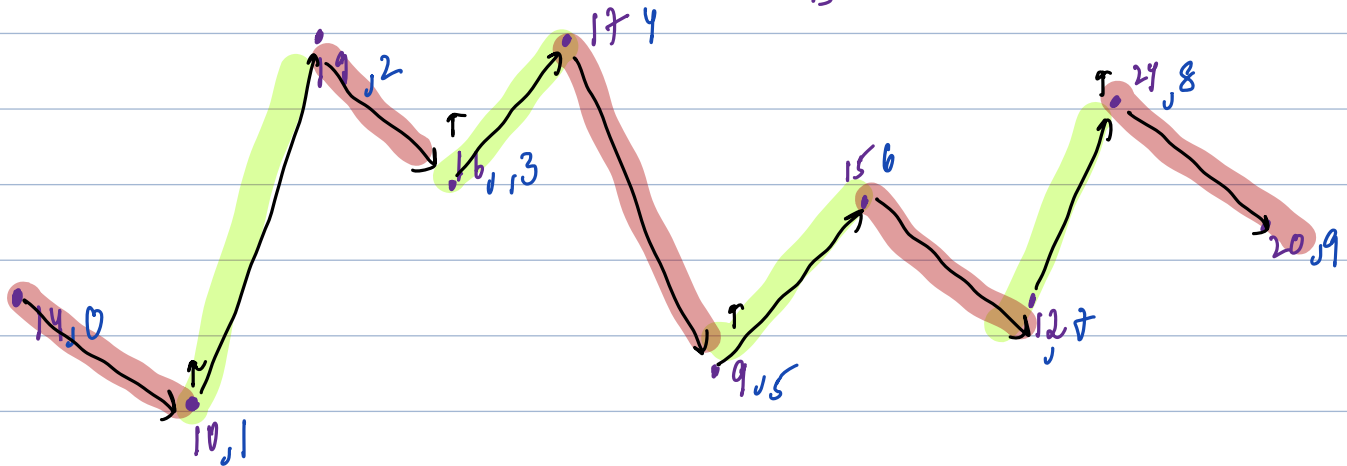


#observations

if $(i \% 2 == 0)$ {
#Ideally $ar[i] > ar[i+1]$

if $(i \% 2 == 1)$ {
#Ideally $ar[i] < ar[i+1]$

Ex2: $ar[] = \{ \overset{0}{14}, \overset{1}{19}, \overset{2}{17}, \overset{3}{19}, \overset{4}{16}, \overset{5}{12}, \overset{6}{15}, \overset{7}{15}, \overset{8}{20}, \overset{9}{24} \}$
 \rightarrow
 $\underset{14}{14}, \underset{10}{10}, \underset{19}{19}, \underset{16}{16}, \underset{19}{19}, \underset{9}{9}, \underset{15}{15}, \underset{12}{12}, \underset{24}{24}, \underset{20}{20}$



int

vector<int> waveArray(vector<int> arr) { TC: $O(N)$ SC: $O(1)$

int n = arr.size();

for(int i = 0; i < n; i++) {

if(i % 2 == 0) {

if(arr[i] < arr[i+1]) { swap arr[i] & arr[i+1] }

else {

if(arr[i] > arr[i+1]) { swap arr[i] & arr[i+1] }

}

}

return arr

3