

In a company there are employees and their efficiency is given in array 'arr' (can be negative). You need to find the maximum efficiency of 3 employees. The efficiency of 3 employees will be calculated by multiplying their individual efficiencies for the given array

Input:

5

3 -2 -8 4 1

Output

64

64

64

3 -2 -8 4 1

3 -2 -8 4 1

$-8 \times -2 = 16$
 $16 \times 4 = 64$

$0 \times 1 \times 0 = 0$

multiply the 2 smallest -ve values
 \times maximum the value

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

max

$-ve \times -ve = +ve$
 $+ve \times +ve = +ve$
 $-ve \times +ve = -ve$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

$-2 \times -1 = 2$
 $2 \times 8 = 16$

Given an array nums with n objects coloured red, white or blue. Sort them in-place so that the same colour are adjacent, with the colours in the order red, white and blue.

We will use the integers 3, 6 and 7 to represent the colour red, white and blue respectively.

You must solve this problem without using the sorting function.

Input:

Nums[] = [3, 6, 3, 7, 6, 3, 7]

Output:

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

DNF

if you have to sort 3 types of value the sort the smallest and the largest so that middle will come automatically to its place

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

3 3 3 6 6 7 7

3 6 3 7 6 3 7

```

    right--;
}
else {
    mid++;
}
}

```

0 1

3	3	3	6	6	7	7
---	---	---	---	---	---	---