

hackerrank.com/challenges/closest-numbers/problem

```
8
9 # Complete the closestNumbers function below.
10 def closestNumbers(arr):
11     #choose any max_num
12     smallest_pairs = []
13     arr.sort()
14     smallest_diff = 1000000
15     for i in range(1, len(arr)):
16         diff = abs(arr[i-1] - arr[i])
17         if diff < smallest_diff:
18             smallest_diff = diff
19             smallest_pairs = [arr[i-1], arr[i]]
20         elif diff == smallest_diff:
21             smallest_pairs.extend([arr[i-1], arr[i]])
22     return smallest_pairs
23
24
25
26 if __name__ == '__main__':
27     fptr = open(os.environ['OUTPUT_PATH'], 'w')
28
29     n = int(raw_input())
30
```

Line: 29 Col: 23

☐ Upload Code as File ☐ Test against custom input

[Run Code](#) [Submit Code](#)

Congratulations!
You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Activate Windows
Go to Settings to activate Windows.

Sample Test case 0

hackerrank.com/challenges/closest-numbers/problem

Change Theme Python 2

```
5 import random
6 import re
7 import sys
8
9 # Complete the closestNumbers function below.
10 def closestNumbers(arr):
11     #choose any max_num
12     smallest_pairs = []
13     arr.sort()
14     smallest_diff = 1000000
15     for i in range(1, len(arr)):
16         diff = abs(arr[i-1] - arr[i])
17         if diff < smallest_diff:
18             smallest_diff = diff
19             smallest_pairs = [arr[i-1], arr[i]]
20         elif diff == smallest_diff:
21             smallest_pairs.extend([arr[i-1], arr[i]])
22     return smallest_pairs
23
24
25
26 if __name__ == '__main__':
27     fptr = open(os.environ['OUTPUT_PATH'], 'w')
28
29     n = int(raw_input())
30
```

Line: 29 Col: 23

Activate Windows
Go to Settings to activate Windows.

hackerrank.com/challenges/closest-numbers/problem

Update

Change Theme Python 2

5 import random

6 import re

7 import sys

8

9 # Complete the closestNumbers function below.

10 def closestNumbers(arr):

11 #choose any max_num

12 smallest_pairs = []

13 arr.sort()

14 smallest_diff = 1000000

15 for i in range(1,len(arr)):

16 diff=abs(arr[i-1] - arr[i])

17 if diff<smallest_diff:

18 smallest_diff = diff

19 smallest_pairs = [arr[i-1], arr[i]]

20 elif diff == smallest_diff:

21 smallest_pairs.extend([arr[i-1], arr[i]])

22 return smallest_pairs

23

24

25

26 if __name__ == '__main__':

27 fptr = open(os.environ['OUTPUT_PATH'], 'w')

28

29 n = int(raw_input())

30

Line: 29 Col: 23

Activate Windows
Go to Settings to activate Windows.

hackerrank.com/challenges/closest-numbers/problem

Practice > Algorithms > Sorting > Closest Numbers

Closest Numbers ☆

Rank: 1688599 | Points: 36/100

64 more points to get your next star!

Your Closest Numbers submission got 35.00 points.

You are now 64 points away from the 2nd star for your problem solving badge.

[Try the next challenge](#) | [Try a Random Challenge](#)

Problem | Submissions | Leaderboard | Discussions | Editorial | Topics

Sorting is useful as the first step in many different tasks. The most common task is to make finding things easier, but there are other uses as well. In this case, it will make it easier to determine which pair or pairs of elements have the smallest absolute difference between them.

Example

$arr = [5, 2, 3, 4, 1]$

Sorted, $arr' = [1, 2, 3, 4, 5]$. Several pairs have the minimum difference of 1: $[(1, 2), (2, 3), (3, 4), (4, 5)]$. Return the array $[1, 2, 2, 3, 3, 4, 4, 5]$.

Note

As shown in the example, pairs may overlap.

Given a list of unsorted integers, arr , find the pair of elements that have the smallest absolute difference between them. If there are multiple pairs, find them all.

Author: HackerRank
Difficulty: Easy
Max Score: 35
Submitted By: 58953

NEED HELP?

[View discussions](#)
[View editorial](#)
[View top submissions](#) to activate Windows.

RESOURCES

[Greedy Technique](#)

```

22 return smallest_pairs
23
24
25
26 if __name__ == '__main__':
27     fptr = open(os.environ['OUTPUT_PATH'], 'w')
28
29     n = int(raw_input())
30

```

Line: 29 Col: 23

☐ Upload Code as File ☐ Test against custom input

[Run Code](#) [Submit Code](#)

You have earned 35.00 points!

You are now 64 points away from the 2nd star for your problem solving badge.

9% 36/100

Congratulations

You solved this challenge. Would you like to challenge your friends?

[Next Challenge](#)

Earn a certificate in Problem Solving

Kudos on your progress! Take the HackerRank Skills Certification test and enrich your profile

[Get Certified](#)

Test case 0

Test case 1

Compiler Message

Success

Activate Windows
Go to Settings to activate Windows.

Username: d.mbugua

My code:

```

def closestNumbers(arr):
    #choose any max_num
    smallest_pairs = []

```

```
arr.sort()
smallest_diff = 1000000
for i in range(1, len(arr)):
    diff = abs(arr[i-1] - arr[i])
    if diff < smallest_diff:
        smallest_diff = diff
        smallest_pairs = [arr[i-1], arr[i]]
    elif diff == smallest_diff:
        smallest_pairs.extend([arr[i-1], arr[i]])
return smallest_pairs

if __name__ == '__main__':
    fptr = open(os.environ['OUTPUT_PATH'], 'w')

    n = int(raw_input())

    arr = map(int, raw_input().rstrip().split())

    result = closestNumbers(arr)

    fptr.write(' '.join(map(str, result)))
    fptr.write('\n')

    fptr.close()
```