Geraldo Braho
COMP 5327.Advanced Algorithms

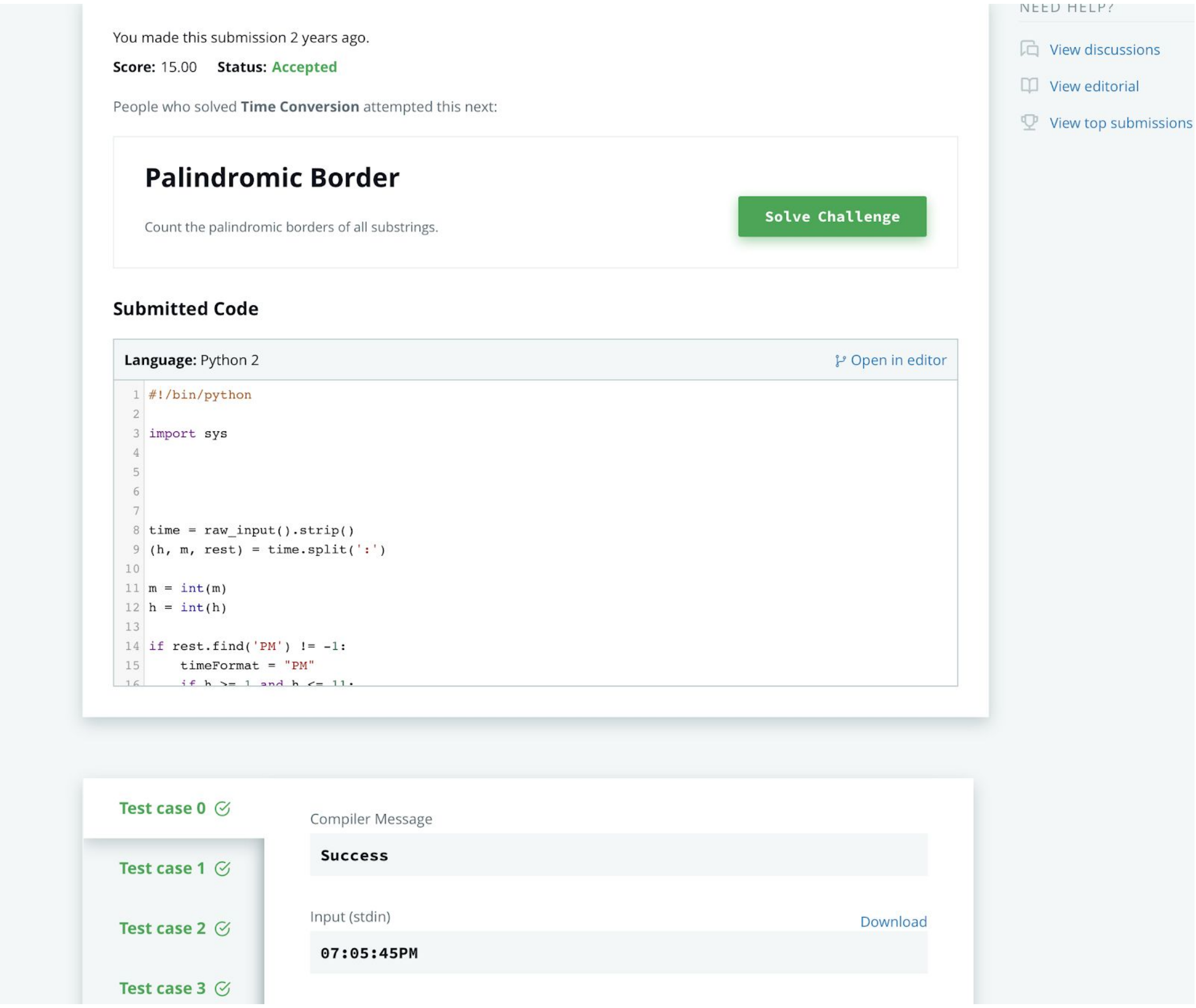
Home-Work 1

Please solve the following questions: https://www.hackerrank.com/challenges/time-conversion/problem And https://www.hackerrank.com/challenges/closest-numbers/problem

1- https://www.hackerrank.com/challenges/time-conversion/problem

Time Conversion

I have solved this question 2 years ago.(https://github.com/geraldo1993/Hackerrank/blob/master/Solutions%20Using%20Python/Algorithm/Warmup/timeConvesion.py)



You made this submission 2 years ago.

Score: 15.00 **Status:** Accepted

People who solved **Time Conversion** attempted this next:

Palindromic Border

Count the palindromic borders of all substrings.

Submitted Code

Language: Python 2

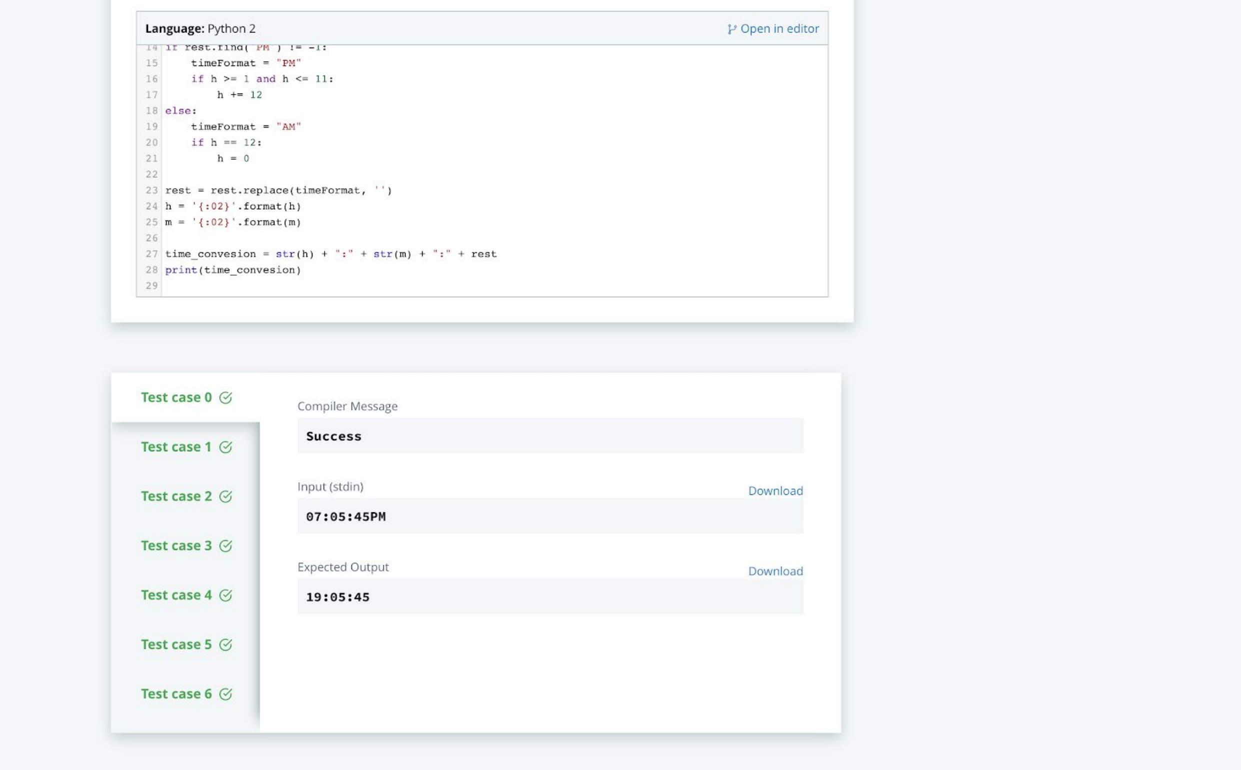
```
#!/bin/python

import sys




time = raw_input().strip()
(h, m, rest) = time.split(':')

m = int(m)
h = int(h)

if rest.find('PM') != -1:
    timeFormat = "PM"
    if h >= 1 and h <= 11:
```

Compiler Message

```
Success
```

Input (stdin)

```
07:05:45PM
```



Language: Python 2

```
if rest.find('PM') != -1:
    timeFormat = "PM"
    if h >= 1 and h <= 11:
        h += 12
else:
    timeFormat = "AM"
    if h == 12:
        h = 0

rest = rest.replace(timeFormat, '')
h = '{:02}'.format(h)
m = '{:02}'.format(m)

time_convesion = str(h) + ":" + str(m) + ":" + rest
print(time_convesion)

```

Compiler Message

```
Success
```

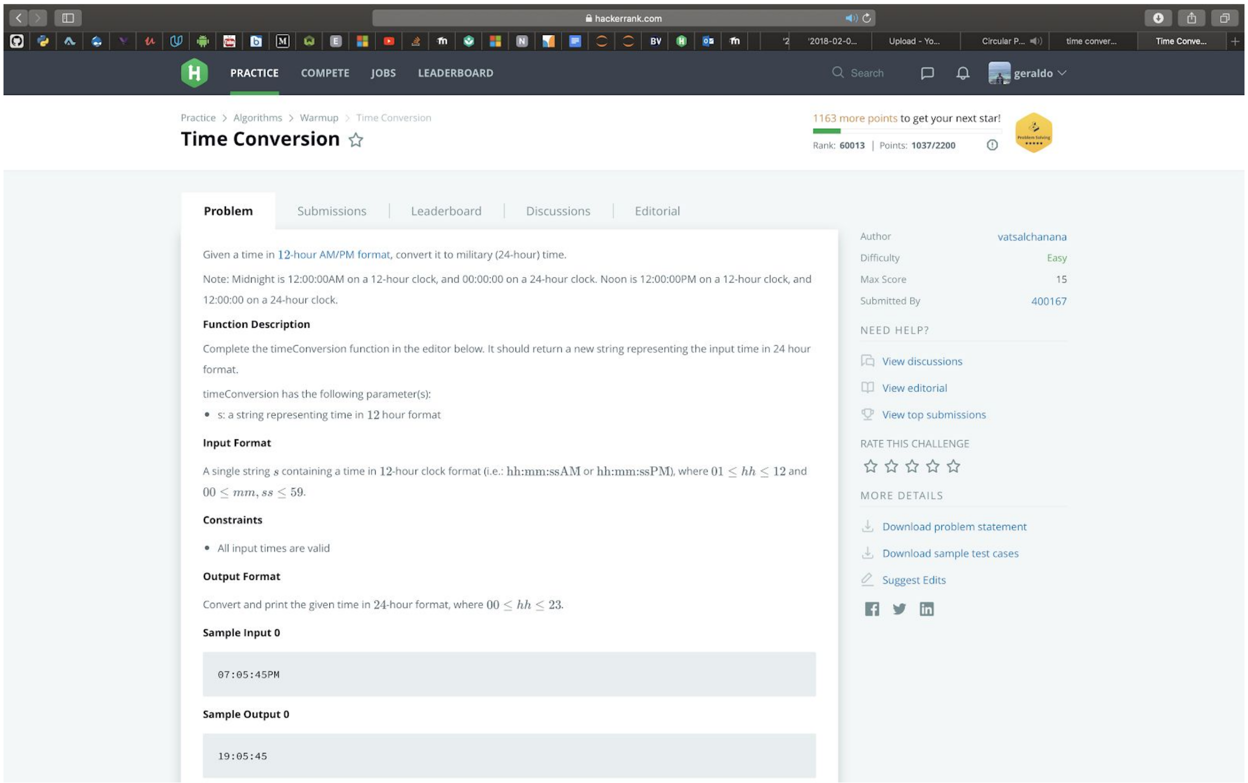
Input (stdin)

```
07:05:45PM
```

Expected Output

```
19:05:45
```

My solution for this class.
https://www.hackerrank.com/challenges/time-conversion/submissions/code/96603164


hackerrank.com
PRACTICE
COMPETE
JOBS
LEADERBOARD
Search
geraldo
Practice > Algorithms > Warmup > Time Conversion
1163 more points to get your next star!
Rank: 60013 | Points: 1037/2200
Problem
Submissions
Leaderboard
Discussions
Editorial

Time Conversion ☆

Author vatsalchanana
Difficulty Easy
Max Score 15
Submitted By 400167

NEED HELP?

- View discussions
- View editorial
- View top submissions

RATE THIS CHALLENGE ☆ ☆ ☆ ☆ ☆

MORE DETAILS

- Download problem statement
- Download sample test cases
- Suggest Edits

Given a time in 12-hour AM/PM format, convert it to military (24-hour) time.

Note: Midnight is 12:00:00AM on a 12-hour clock, and 00:00:00 on a 24-hour clock. Noon is 12:00:00PM on a 12-hour clock, and 12:00:00 on a 24-hour clock.

Function Description

Complete the timeConversion function in the editor below. It should return a new string representing the input time in 24 hour format.

timeConversion has the following parameter(s):

- s: a string representing time in 12 hour format

Input Format

A single string s containing a time in 12-hour clock format (i.e.: $\mathrm{hh{:}mm{:}ssAM}$ or $\mathrm{hh{:}mm{:}ssPM}$), where $01 \le hh \le 12$ and $00 \le mm, ss \le 59$.

Constraints

- All input times are valid

Output Format

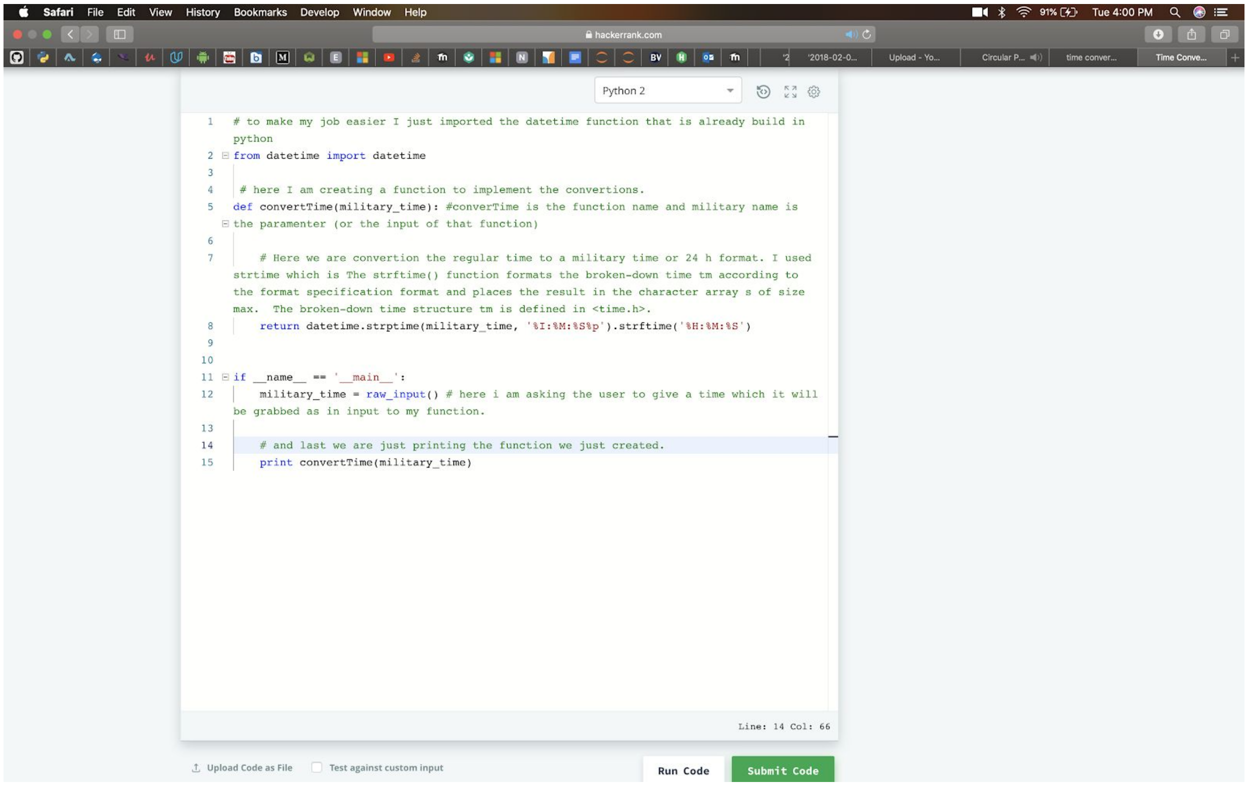
Convert and print the given time in 24-hour format, where $00 \le hh \le 23$.

Sample Input 0

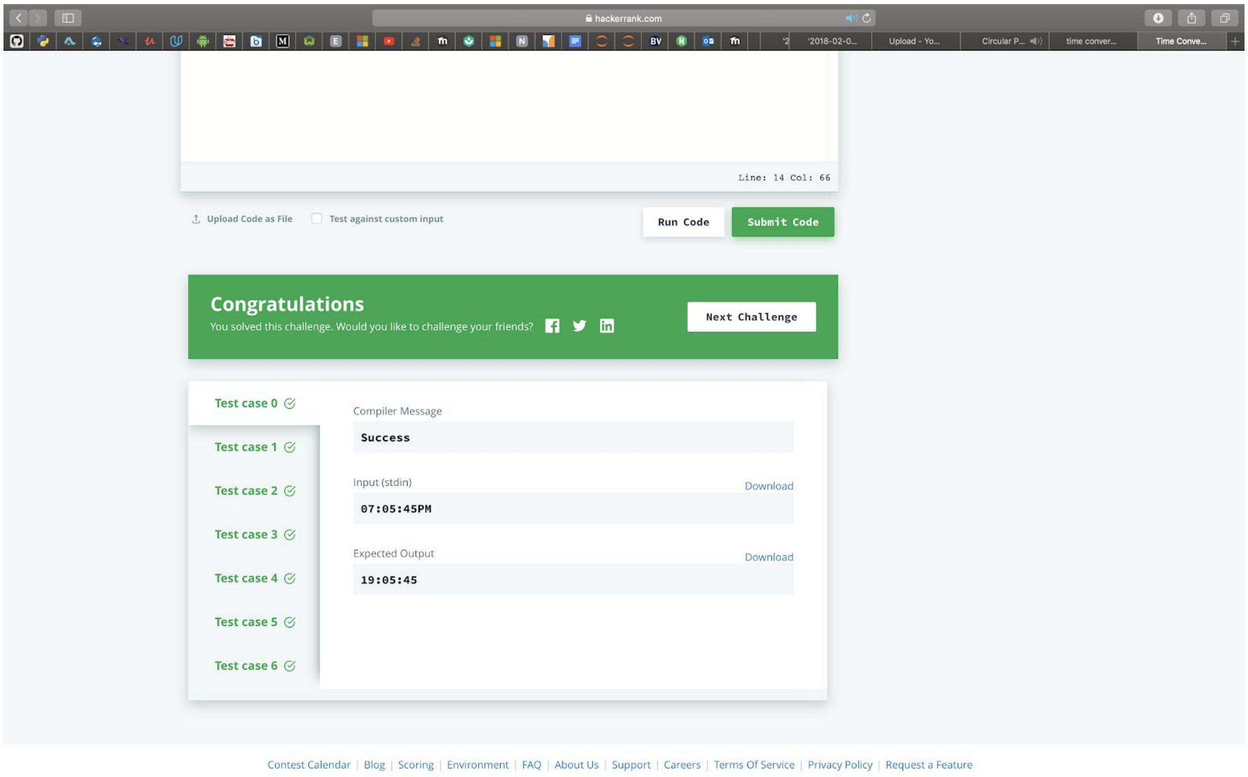
```
07:05:45PM
```

Sample Output 0

```
19:05:45
```



```python
# to make my job easier I just imported the datetime function that is already build in python
from datetime import datetime

 # here I am creating a function to implement the convertions.
def convertTime(military_time): #converTime is the function name and military name is the paramenter (or the input of that function)

    # Here we are convertion the regular time to a military time or 24 h format. I used strtime which is The strftime() function formats the broken-down time tm according to the format specification format and places the result in the character array s of size max.  The broken-down time structure tm is defined in <time.h>.
    return datetime.strptime(military_time, '%I:%M:%S%p').strftime('%H:%M:%S')


if __name__ == '__main__':
    military_time = raw_input() # here i am asking the user to give a time which it will be grabbed as in input to my function.

    # and last we are just printing the function we just created.
    print convertTime(military_time)
```



Congratulations

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

Compiler Message

```
Success
```

Input (stdin)

```
07:05:45PM
```

Expected Output

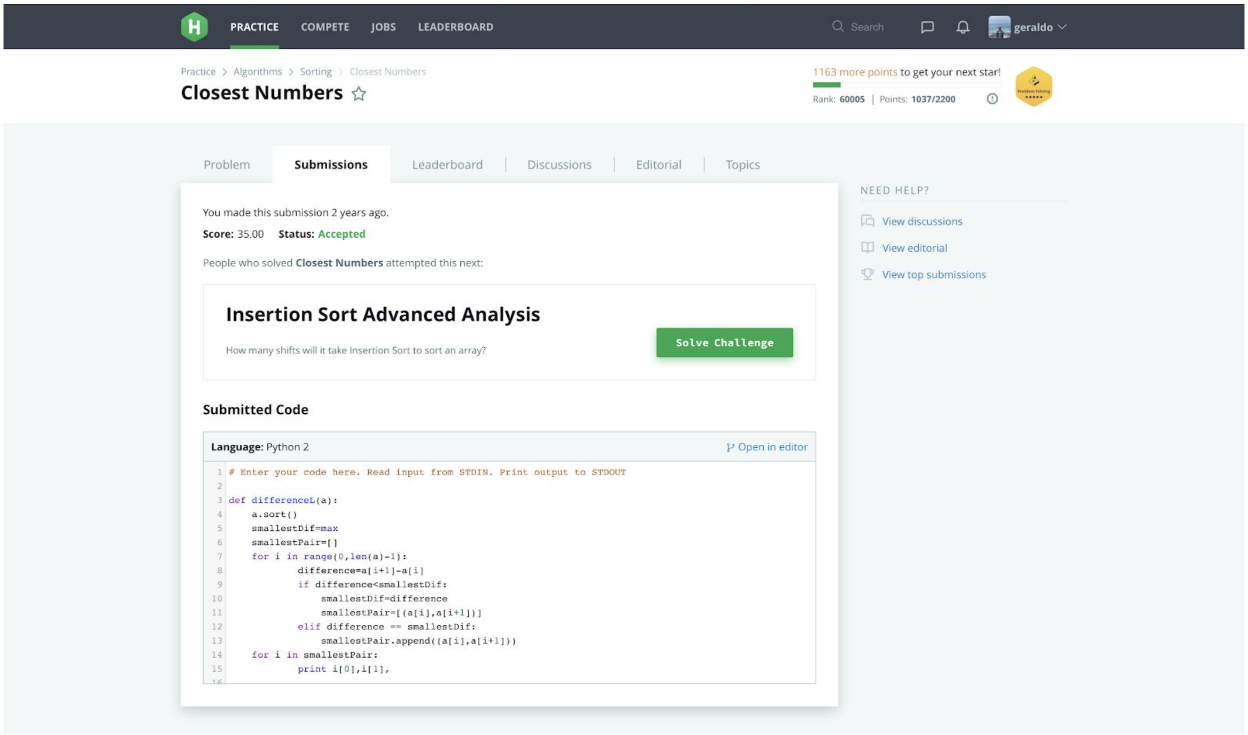
```
19:05:45
```

Closest Numbers

2- https://www.hackerrank.com/challenges/closest-numbers/problem

I had solve this question 2 years ago as well and this time I will try to optimize it and pass all the cases.
My solution 2 years ago.
https://github.com/geraldo1993/Hackerrank/blob/master/Solutions%20Using%20Python/Algorith

m/Sorting/Closest%20Numbers.py (You may check my github as well).



Practice > Algorithms > Sorting > Closest Numbers

Closest Numbers ☆

1163 more points to get your next star!

Rank: 60005 | Points: 1037/2200

You made this submission 2 years ago.

Score: 35.00 **Status:** Accepted

People who solved **Closest Numbers** attempted this next:

Insertion Sort Advanced Analysis

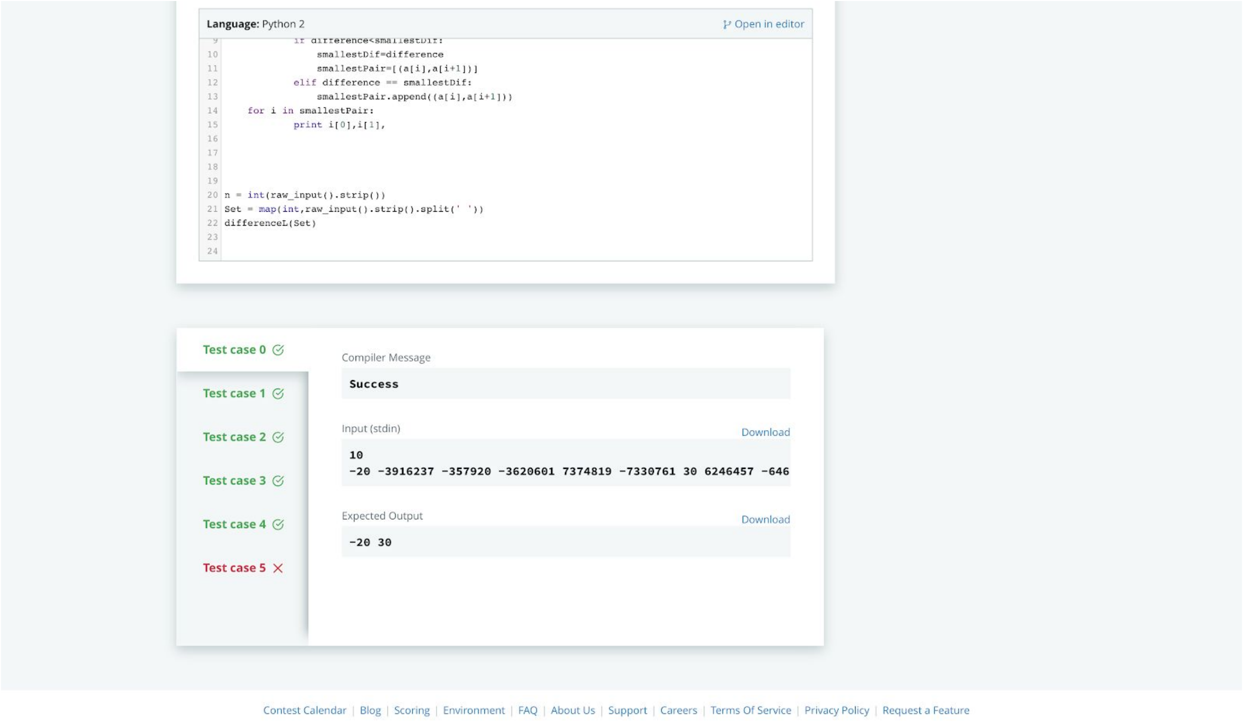
How many shifts will it take Insertion Sort to sort an array?

NEED HELP?

Submitted Code

Language: Python 2

```
# Enter your code here. Read input from STDIN. Print output to STDOUT

def differenceL(a):
    a.sort()
    smallestDif=max
    smallestPair=[]
    for i in range(0,len(a)-1):
            difference=a[i+1]-a[i]
            if difference<smallestDif:
                smallestDif=difference
                smallestPair=[(a[i],a[i+1])]
            elif difference == smallestDif:
                smallestPair.append((a[i],a[i+1]))
    for i in smallestPair:
            print i[0],i[1],
```



Language: Python 2

```
            if difference<smallestDif:
                smallestDif=difference
                smallestPair=[(a[i],a[i+1])]
            elif difference == smallestDif:
                smallestPair.append((a[i],a[i+1]))
    for i in smallestPair:
            print i[0],i[1],




n = int(raw_input().strip())
Set = map(int,raw_input().strip().split(' '))
differenceL(Set)


```

Test case 0 ✓

Test case 1 ✓

Test case 2 ✓

Test case 3 ✓

Test case 4 ✓

Test case 5 ✗

Compiler Message

```
Success
```

Input (stdin) Download

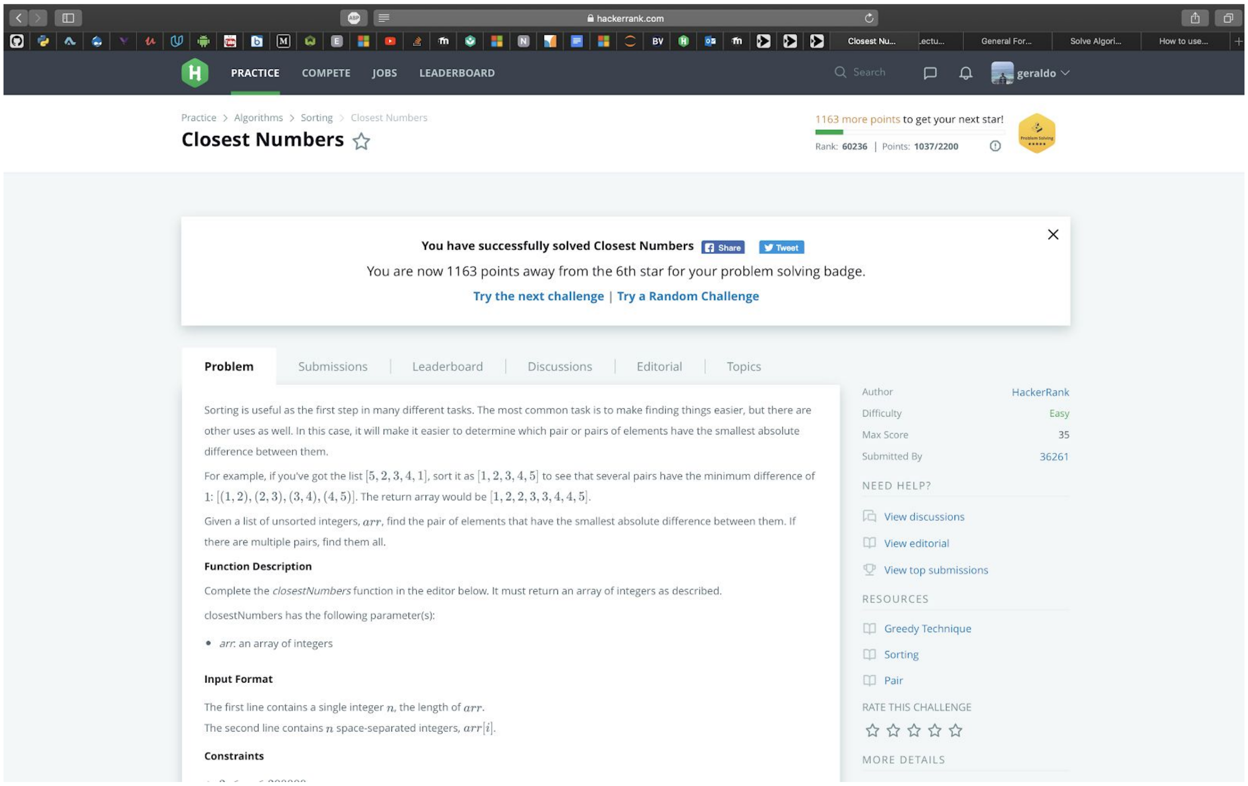
```
10
-20 -3916237 -357920 -3620601 7374819 -7330761 30 6246457 -646
```

Expected Output Download

```
-20 30
```

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature

My solution now that I am taking advanced algorithm class.



Practice > Algorithms > Sorting > Closest Numbers

Closest Numbers ☆

1163 more points to get your next star!

Rank: 60236 | Points: 1037/2200

You have successfully solved Closest Numbers Share Tweet

You are now 1163 points away from the 6th star for your problem solving badge.

Try the next challenge | **Try a Random Challenge**

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.

For example, if you've got the list $[5, 2, 3, 4, 1]$, sort it as $[1, 2, 3, 4, 5]$ to see that several pairs have the minimum difference of 1: $[(1, 2), (2, 3), (3, 4), (4, 5)]$. The return array would be $[1, 2, 2, 3, 3, 4, 4, 5]$.

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.

Function Description

Complete the *closestNumbers* function in the editor below. It must return an array of integers as described.

closestNumbers has the following parameter(s):

- *arr*: an array of integers

Input Format

The first line contains a single integer n, the length of arr.
The second line contains n space-separated integers, $arr[i]$.

Constraints

Author	HackerRank
Difficulty	Easy
Max Score	35
Submitted By	36261

NEED HELP?

- View discussions
- View editorial
- View top submissions

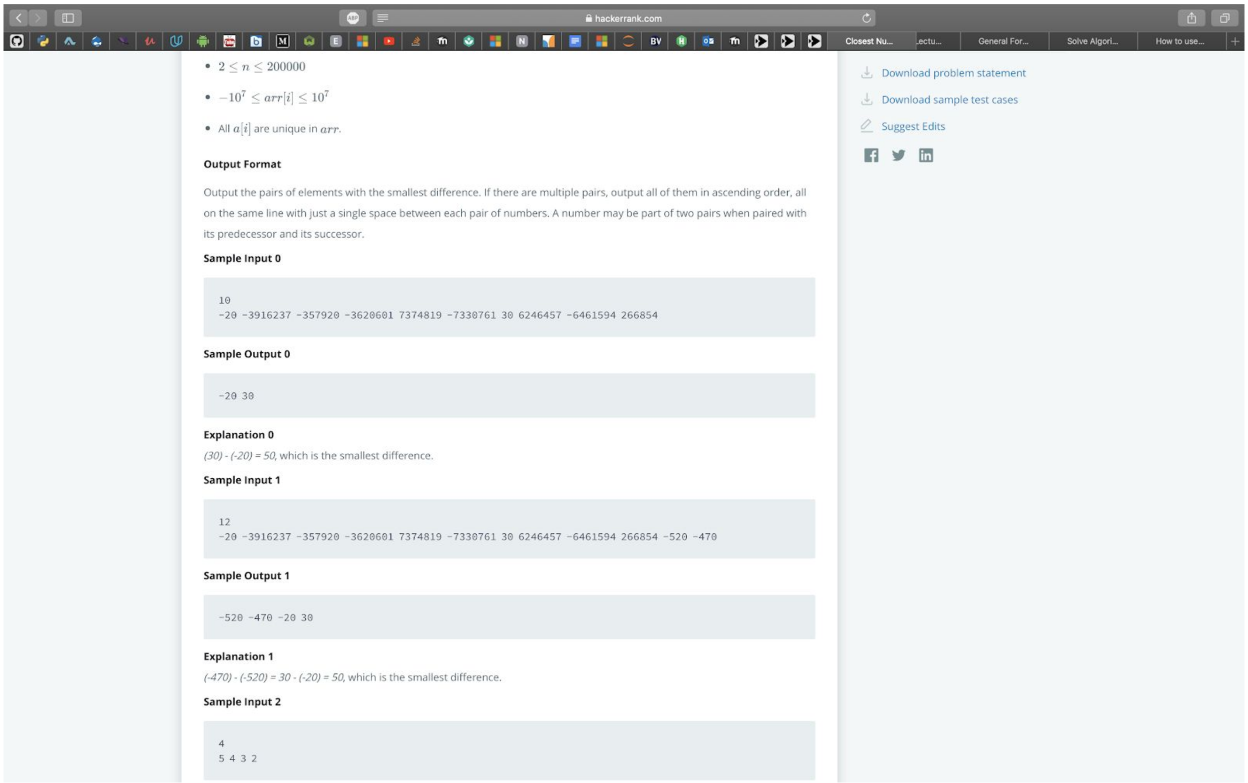
RESOURCES

- Greedy Technique
- Sorting
- Pair

RATE THIS CHALLENGE

☆ ☆ ☆ ☆ ☆

MORE DETAILS



- $2 \le n \le 200000$
- $-10^7 \le arr[i] \le 10^7$
- All $a[i]$ are unique in arr.

Output Format

Output the pairs of elements with the smallest difference. If there are multiple pairs, output all of them in ascending order, all on the same line with just a single space between each pair of numbers. A number may be part of two pairs when paired with its predecessor and its successor.

Sample Input 0

```
10
-20 -3916237 -357920 -3620601 7374819 -7330761 30 6246457 -6461594 266854
```

Sample Output 0

```
-20 30
```

Explanation 0

(30) - (-20) = 50, which is the smallest difference.

Sample Input 1

```
12
-20 -3916237 -357920 -3620601 7374819 -7330761 30 6246457 -6461594 266854 -520 -470
```

Sample Output 1

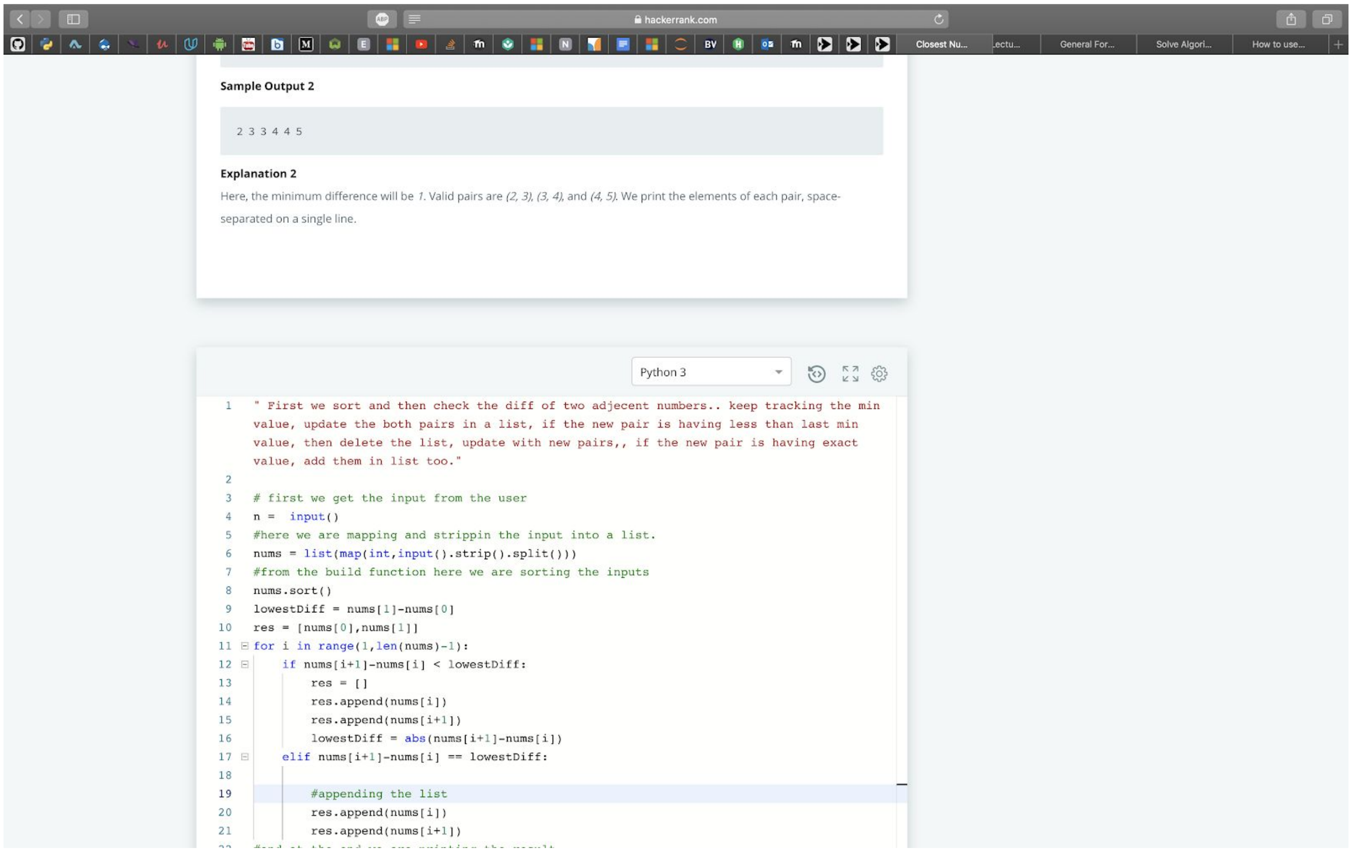
```
-520 -470 -20 30
```

Explanation 1

(-470) - (-520) = 30 - (-20) = 50, which is the smallest difference.

Sample Input 2

```
4
5 4 3 2
```



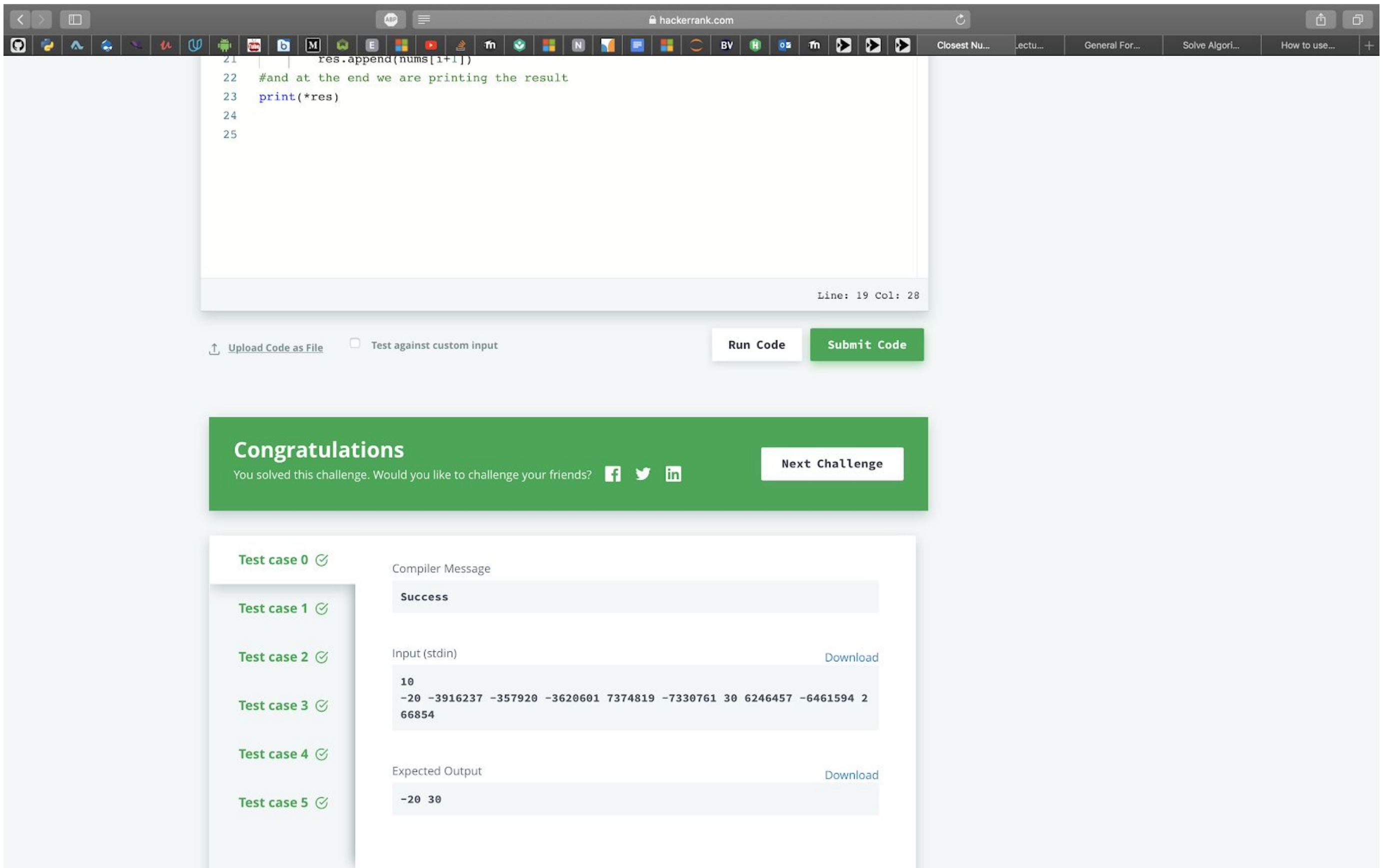
Sample Output 2

```
2 3 3 4 4 5
```

Explanation 2

Here, the minimum difference will be *1*. Valid pairs are *(2, 3)*, *(3, 4)*, and *(4, 5)*. We print the elements of each pair, space-separated on a single line.

```python
" First we sort and then check the diff of two adjecent numbers.. keep tracking the min value, update the both pairs in a list, if the new pair is having less than last min value, then delete the list, update with new pairs,, if the new pair is having exact value, add them in list too."

# first we get the input from the user
n =  input()
#here we are mapping and strippin the input into a list.
nums = list(map(int,input().strip().split()))
#from the build function here we are sorting the inputs
nums.sort()
lowestDiff = nums[1]-nums[0]
res = [nums[0],nums[1]]
for i in range(1,len(nums)-1):
    if nums[i+1]-nums[i] < lowestDiff:
        res = []
        res.append(nums[i])
        res.append(nums[i+1])
        lowestDiff = abs(nums[i+1]-nums[i])
    elif nums[i+1]-nums[i] == lowestDiff:

        #appending the list
        res.append(nums[i])
        res.append(nums[i+1])
#and at the end we are printing the result
```



```
            res.append(nums[i+1])
#and at the end we are printing the result
print(*res)
```

Congratulations

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

Compiler Message

```
Success
```

Input (stdin) Download

```
10
-20 -3916237 -357920 -3620601 7374819 -7330761 30 6246457 -6461594 2
66854
```

Expected Output Download

```
-20 30
```