

Цель: создание аналога сайта Hackerrank. Основные элементы пользовательского интерфейса: авторизация, выбор задач по различным критериям (сложность, тематика), предоставление решения в виде файла исходного кода, просмотр результатов тестирования предложенного решения. Также: просмотр статистики, возможность участия в обсуждении задачи, возможность предлагать собственные задачи к публикации, наличие привилегированных пользователей, способных публиковать задачи.

[Algorithms](#) | [Data Structures](#)

**Arrays - DS**  
Easy, Max Score: 10, Success Rate: 93.94%,  
Accessing and using arrays.

Solve Challenge

**Dynamic Array**  
Easy, Max Score: 15, Success Rate: 83.05%,

Solve Challenge

**Left Rotation**  
Easy, Max Score: 20, Success Rate: 87.25%,

Solve Challenge

**Sparse Arrays**  
Medium, Max Score: 25, Success Rate: 96.62%,

Solve Challenge

**Get an interview with Booking.com**  
Apply and solve the code challenges to land an interview with Booking.com

View

**STATUS**  
☐ Solved  
☒ Unsolved

**DIFFICULTY**  
☐ Easy  
☐ Medium  
☐ Hard

**SUBDOMAINS**  
☐ Arrays  
☐ Linked Lists  
☐ Trees  
☐ Balanced Trees  
☐ Stacks  
☐ Queues  
☐ Heap  
☐ Disjoint Set  
☐ Multiple Choice  
☐ Trie  
☐ Advanced

**Problem** | [Submissions](#) | [Leaderboard](#) | [Discussions](#) | [Editorial](#)

Given a  $6 \times 6$  2D Array, *arr*:

```
1 1 1 0 0 0
0 1 0 0 0 0
1 1 1 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
```

We define an hourglass in *A* to be a subset of values with indices falling in this pattern in *arr*'s graphical representation:

```
a b c
  d
e f g
```

There are 16 hourglasses in *arr*, and an *hourglass sum* is the sum of an hourglass' values. Calculate the hourglass sum for every hourglass in *arr*, then print the *maximum* hourglass sum.

For example, given the 2D array:

```
-9 -9 -9 1 1 1
0 -9 0 4 3 2
-9 -9 -9 1 2 3
0 0 8 6 6 0
0 0 0 -2 0 0
0 0 1 2 4 0
```

We calculate the following 16 hourglass values:

```
-63, -34, -9, 12,
-10, 0, 28, 23,
-27, -11, -2, 10,
9, 17, 25, 18
```

Our highest hourglass value is 28 from the hourglass:

**Author** [Shafaet](#)  
**Difficulty** [Easy](#)  
**Max Score** 15  
**Submitted By** 197712

**NEED HELP?**  
[View discussions](#)  
[View editorial](#)  
[View top submissions](#)

**RATE THIS CHALLENGE**  
☆☆☆☆

**MORE DETAILS**  
[Download problem statement](#)  
[Download sample test cases](#)  
[Suggest Edits](#)  
[f](#) [t](#) [in](#)

```


31 for _ in range(6):
32     arr.append(list(map(int, input().rstrip().split())))
33
34 result = hourglassSum(arr)
35
36 fptr.write(str(result) + '\n')
37
38 fptr.close()
39

```

Line: 22 Col: 41

☒ Upload Code as File ☐ Test against custom input

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

 You have earned 15.00 points!  
These points will also count towards your progress in the Problem Solving Badge.

35% 135/200

## Congratulations

You solved this challenge. Would you like to challenge your friends? [f](#) [t](#) [in](#) [Next Challenge](#)

[Testcase 0](#) | [Testcase 1](#) | [Testcase 2](#) | [Testcase 8](#) | [9 Testcases](#) ▾

Input (stdin)	Download	Expected Output	Download
<pre> -9 -9 -9 1 1 1 0 -9 0 4 3 2 -9 -9 -9 1 2 3 0 0 0 0 0 0 </pre>		28	

Compiler Message

Success

Детали реализации:

- \*) язык программирования - Python
- \*) написание бекенд-сервера — aiohttp
- \*) шаблонизатор — jinja2
- \*) верстка CSS — twitter bootstrap
- \*) база данных — MongoDB
- \*) использование готовых инструментов для компиляции/интерпретации — gcc и т.д.
- \*) компиляция/интерпретация и тестирование в отдельных процессах
- \*) использование docker для выполнения кода в docker-контейнере с целью защиты от вредоносного кода
- \*) использование nginx для отдачи статического контента (файлы CSS-стилей, JS-скрипты), для кеширования некоторых запросов (например, статистики), для распределения нагрузки между бекенд-серверами

Открытый вопрос — математическая модель (аналитическая часть) данной темы.