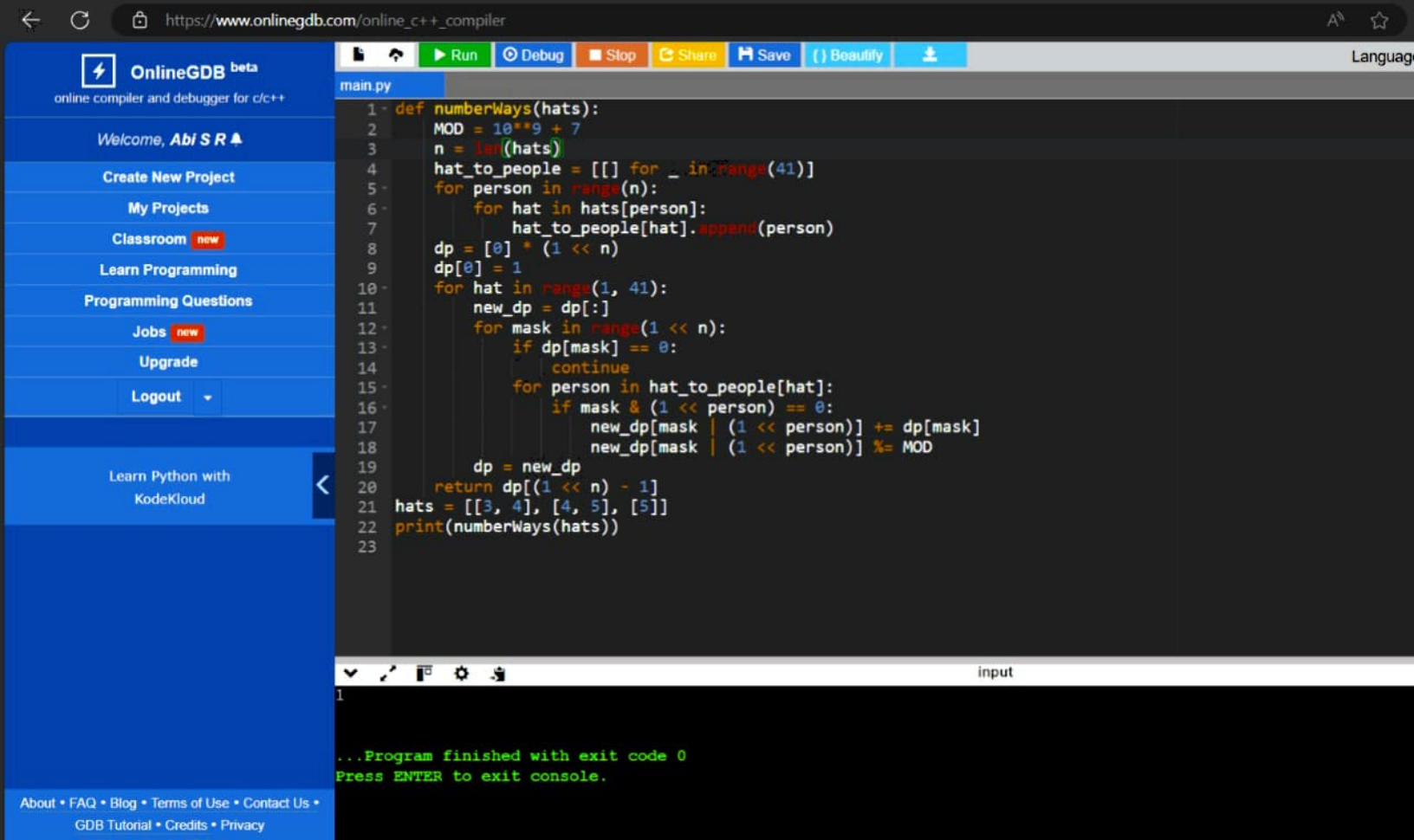
Run | Debug | Stop | Share | Save | { } Beautify

main.py

```python
def destCity(paths):
    start_cities = set()
    end_cities = set()

    for path in paths:
        start_cities.add(path[0])
        end_cities.add(path[1])

    return (end_cities - start_cities).pop()

paths = [["London","New York"],["New York","Lima"],["Lima","Sao Paulo"]]
print(destCity(paths))
```

input

```
Sao Paulo


...Program finished with exit code 0
Press ENTER to exit console.
```


https://www.onlinegdb.com/online_c++_compiler
OnlineGDB beta
online compiler and debugger for c/c++
Welcome, Abi S R
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Jobs new
Upgrade
Logout
Learn Python with KodeKloud
Run
Debug
Stop
Share
Save
{ } Beautify
Language
main.py
def numberWays(hats):
MOD = 10**9 + 7
n = len(hats)
hat_to_people = [[] for _ in range(41)]
for person in range(n):
for hat in hats[person]:
hat_to_people[hat].append(person)
dp = [0] * (1 << n)
dp[0] = 1
for hat in range(1, 41):
new_dp = dp[:]
for mask in range(1 << n):
if dp[mask] == 0:
continue
for person in hat_to_people[hat]:
if mask & (1 << person) == 0:
new_dp[mask | (1 << person)] += dp[mask]
new_dp[mask | (1 << person)] %= MOD
dp = new_dp
return dp[(1 << n) - 1]
hats = [[3, 4], [4, 5], [5]]
print(numberWays(hats))
input
1
...Program finished with exit code 0
Press ENTER to exit console.
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy

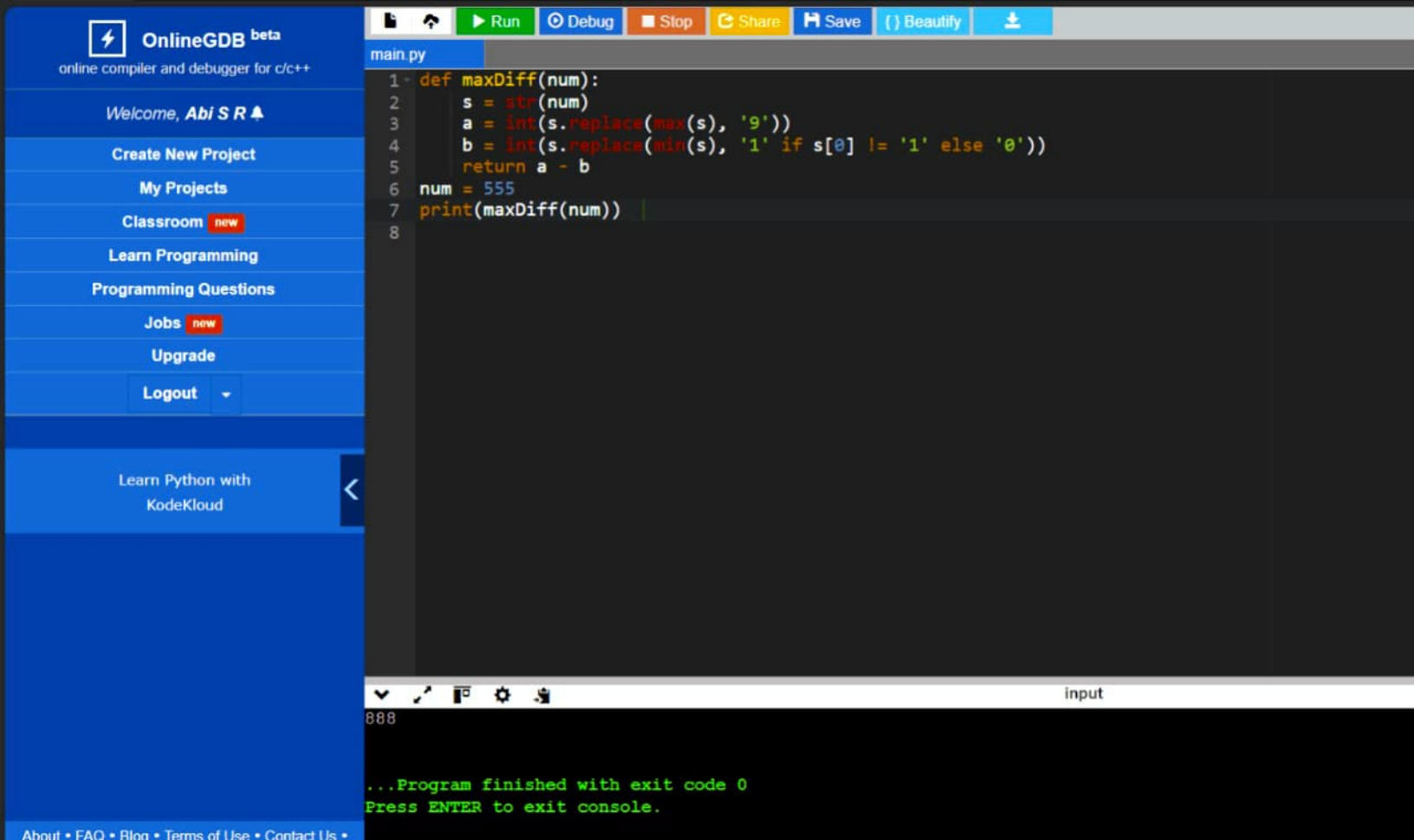
Run | Debug | Stop | Share | Save | { } Beautify

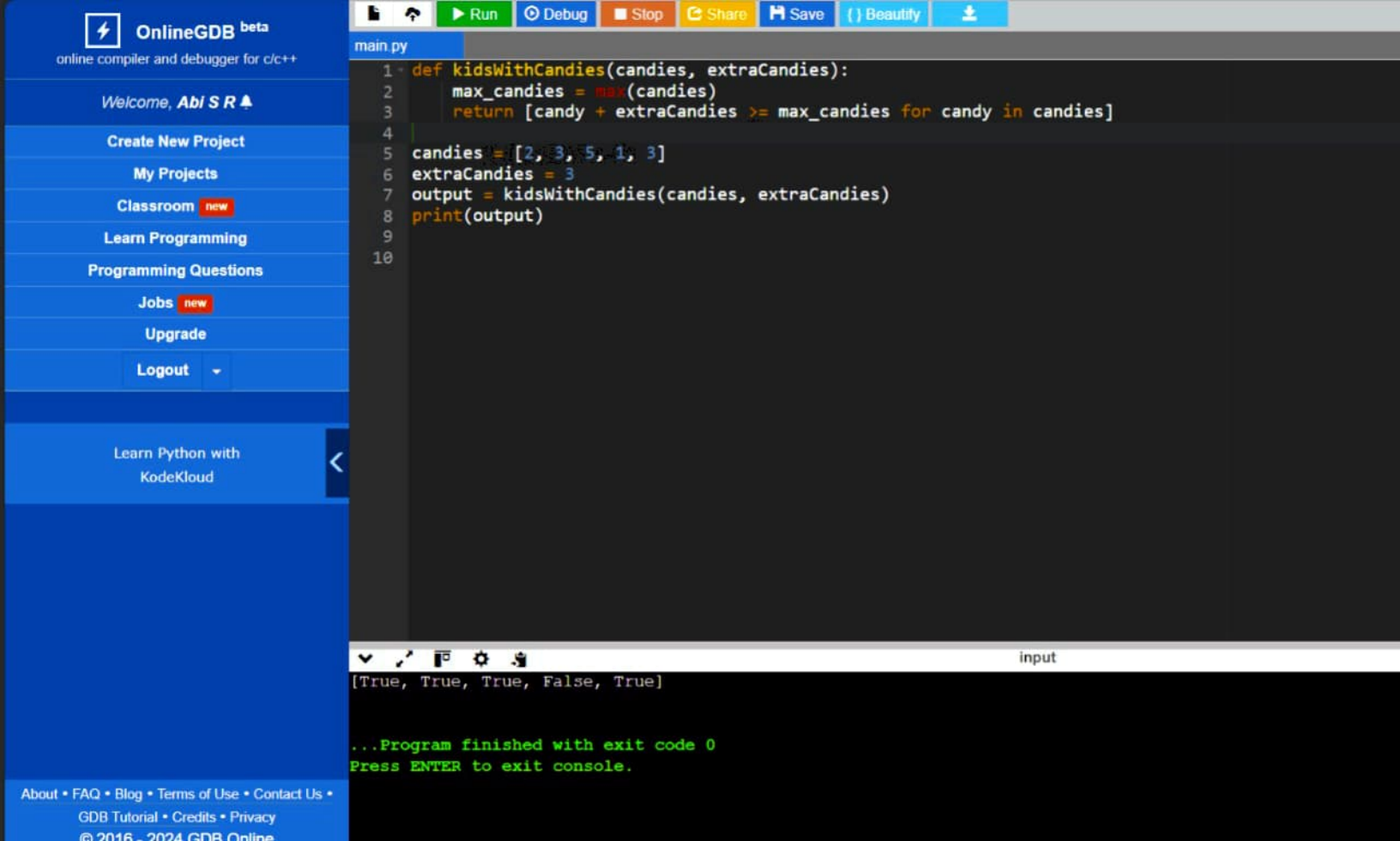
main.py

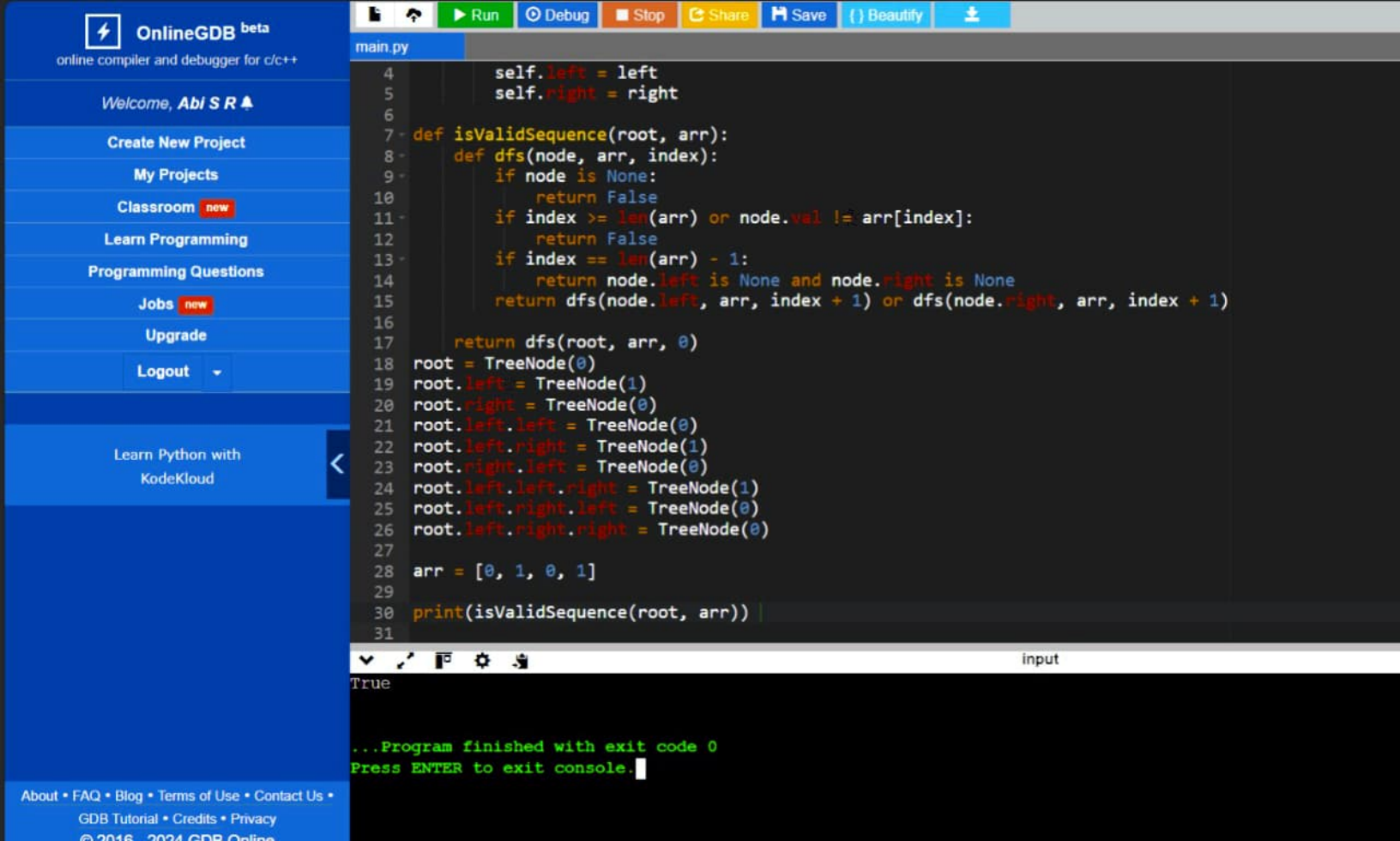
```python
def checkIfCanBreak(s1, s2):
    return all(x >= y for x, y in zip(sorted(s1), sorted(s2)))

s1 = "abc"
s2 = "xya"
print(checkIfCanBreak(s1, s2))

```

input

```
False


...Program finished with exit code 0
Press ENTER to exit console.
```


OnlineGDB beta
online compiler and debugger for c/c++
Welcome, Abi S R
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Jobs new
Upgrade
Logout
Learn Python with
KodeKloud
About • FAQ • Blog • Terms of Use • Contact Us •
Run
Debug
Stop
Share
Save
{ } Beautify
main.py
1 def maxDiff(num):
2 s = str(num)
3 a = int(s.replace(max(s), '9'))
4 b = int(s.replace(min(s), '1' if s[0] != '1' else '0'))
5 return a - b
6 num = 555
7 print(maxDiff(num))
8
input
888
...Program finished with exit code 0
Press ENTER to exit console.


OnlineGDB beta
online compiler and debugger for c/c++
Welcome, Abi S R
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Jobs new
Upgrade
Logout
Learn Python with
KodeKloud
Run
Debug
Stop
Share
Save
{ } Beautify
main.py
1 def kidsWithCandies(candies, extraCandies):
2 max_candies = max(candies)
3 return [candy + extraCandies >= max_candies for candy in candies]
4
5 candies = [2, 3, 5, 1, 3]
6 extraCandies = 3
7 output = kidsWithCandies(candies, extraCandies)
8 print(output)
9
10
input
[True, True, True, False, True]
...Program finished with exit code 0
Press ENTER to exit console.
About • FAQ • Blog • Terms of Use • Contact Us •
GDB Tutorial • Credits • Privacy
© 2016 - 2024 GDB Online



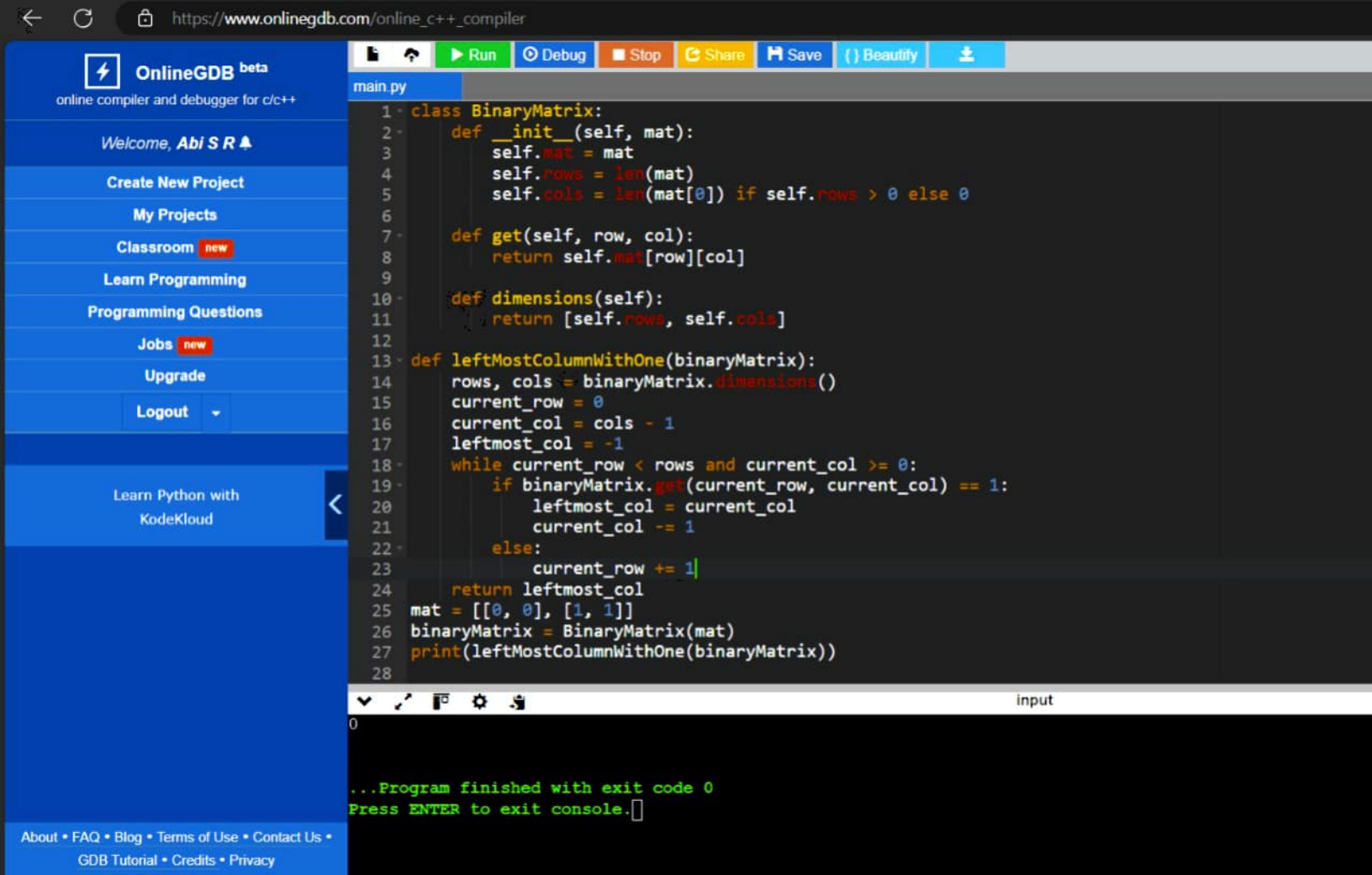
Run | Debug | Stop | Share | Save | { } Beautify | Language Python 3

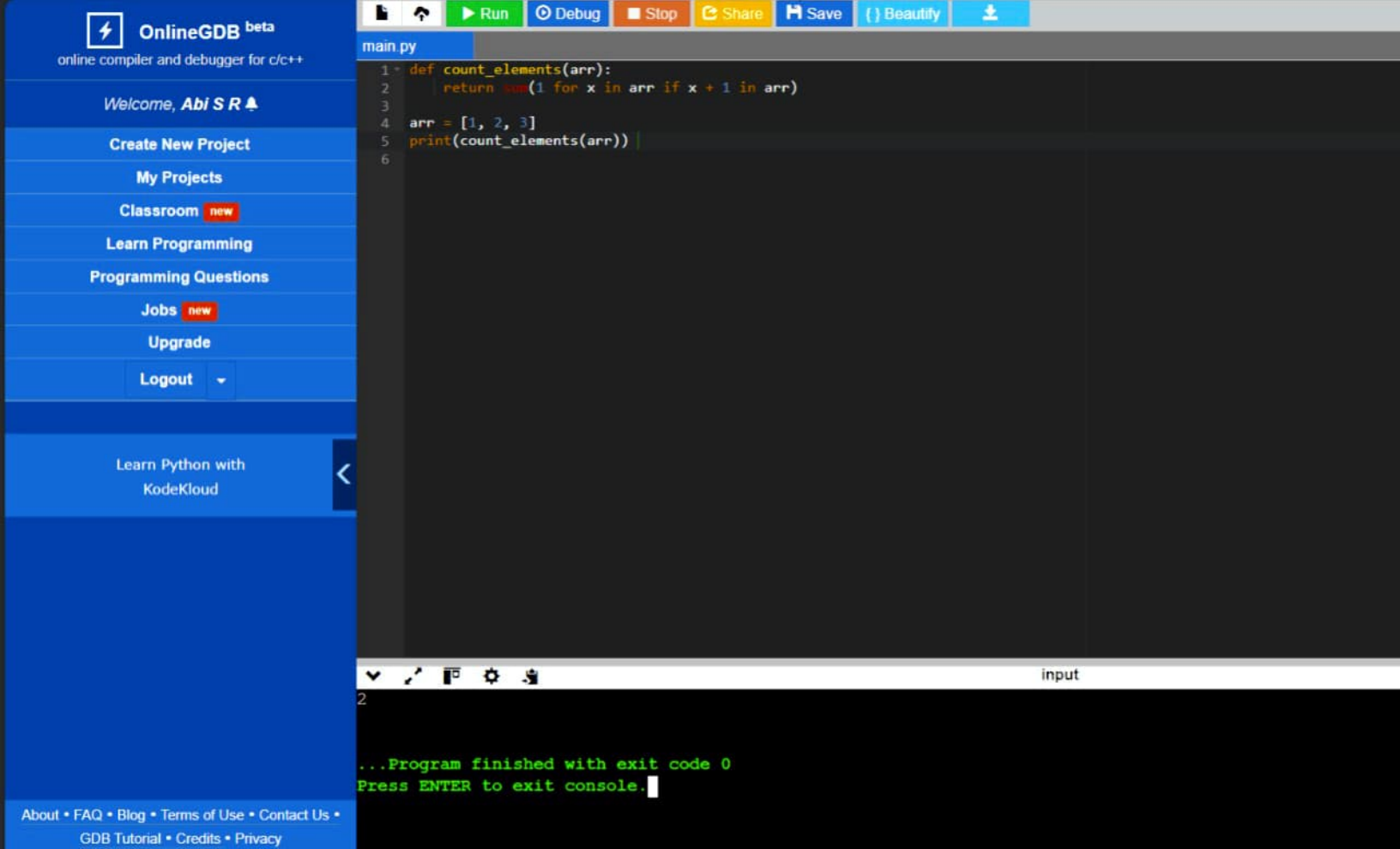
main.py

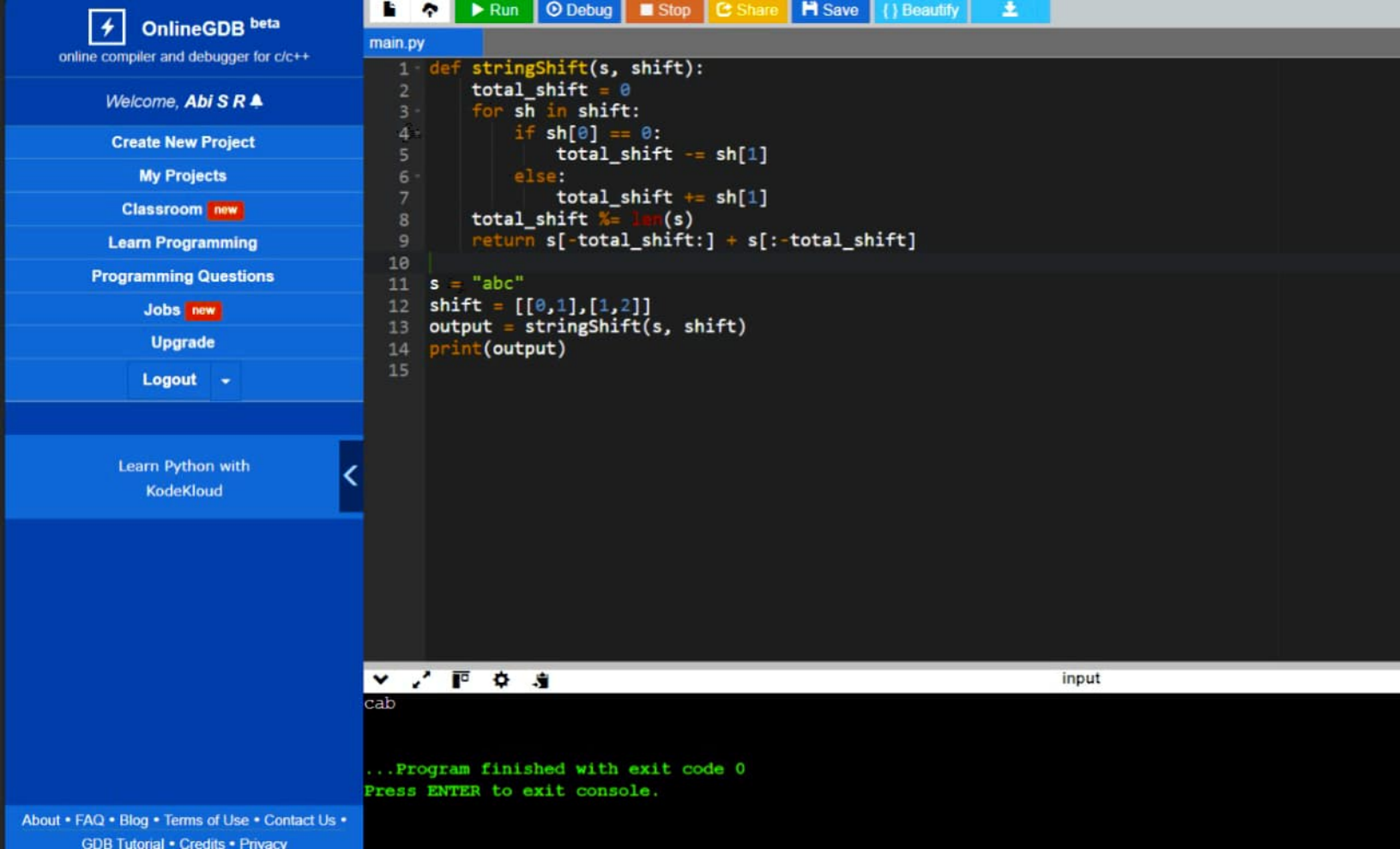
```python
class FirstUnique:
    def __init__(self, nums):
        self.queue = deque()
        self.count = defaultdict(int)
        for num in nums:
            self.add(num)
    def showFirstUnique(self):
        while self.queue and self.count[self.queue[0]] > 1:
            self.queue.popleft()
        if self.queue:
            return self.queue[0]
        else:
            return -1
    def add(self, value):
        self.queue.append(value)
        self.count[value] += 1
commands = ["FirstUnique", "showFirstUnique", "add", "showFirstUnique", "add", "showFirstUnique", "add", "showFirstUnique"]
arguments = [[[2, 3, 5]], [], [5], [], [2], [], [3], []]

firstUnique = FirstUnique(arguments[0][0])
output = [None]
for i in range(1, len(commands)):
    if commands[i] == "showFirstUnique":
        output.append(firstUnique.showFirstUnique())
    elif commands[i] == "add":
        firstUnique.add(arguments[i][0])
        output.append(None)
print(output)
```

input

```
[None, 2, None, 2, None, 3, None, -1]


...Program finished with exit code 0
Press ENTER to exit console.
```


https://www.onlinegdb.com/online_c++_compiler
OnlineGDB beta
online compiler and debugger for c/c++
Welcome, Abi S R
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Jobs new
Upgrade
Logout
Learn Python with KodeKloud
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
Run
Debug
Stop
Share
Save
{ } Beautify
main.py
class BinaryMatrix:
def __init__(self, mat):
self.mat = mat
self.rows = len(mat)
self.cols = len(mat[0]) if self.rows > 0 else 0
def get(self, row, col):
return self.mat[row][col]
def dimensions(self):
return [self.rows, self.cols]
def leftMostColumnWithOne(binaryMatrix):
rows, cols = binaryMatrix.dimensions()
current_row = 0
current_col = cols - 1
leftmost_col = -1
while current_row < rows and current_col >= 0:
if binaryMatrix.get(current_row, current_col) == 1:
leftmost_col = current_col
current_col -= 1
else:
current_row += 1
return leftmost_col
mat = [[0, 0], [1, 1]]
binaryMatrix = BinaryMatrix(mat)
print(leftMostColumnWithOne(binaryMatrix))
input
0
...Program finished with exit code 0
Press ENTER to exit console.


OnlineGDB beta
online compiler and debugger for c/c++
Welcome, Abi S R
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Jobs new
Upgrade
Logout
Learn Python with
KodeKloud
About • FAQ • Blog • Terms of Use • Contact Us •
GDB Tutorial • Credits • Privacy
Run
Debug
Stop
Share
Save
{ } Beautify
main.py
1 def count_elements(arr):
2 return sum(1 for x in arr if x + 1 in arr)
3
4 arr = [1, 2, 3]
5 print(count_elements(arr))
6
input
2
...Program finished with exit code 0
Press ENTER to exit console.


OnlineGDB beta
online compiler and debugger for c/c++
Welcome, Abi S R
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Jobs new
Upgrade
Logout
Learn Python with
KodeKloud
About • FAQ • Blog • Terms of Use • Contact Us •
GDB Tutorial • Credits • Privacy
Run
Debug
Stop
Share
Save
{ } Beautify
main.py
1 def stringShift(s, shift):
2 total_shift = 0
3 for sh in shift:
4 if sh[0] == 0:
5 total_shift -= sh[1]
6 else:
7 total_shift += sh[1]
8 total_shift %= len(s)
9 return s[-total_shift:] + s[:-total_shift]
10
11 s = "abc"
12 shift = [[0,1],[1,2]]
13 output = stringShift(s, shift)
14 print(output)
15
input
cab
...Program finished with exit code 0
Press ENTER to exit console.