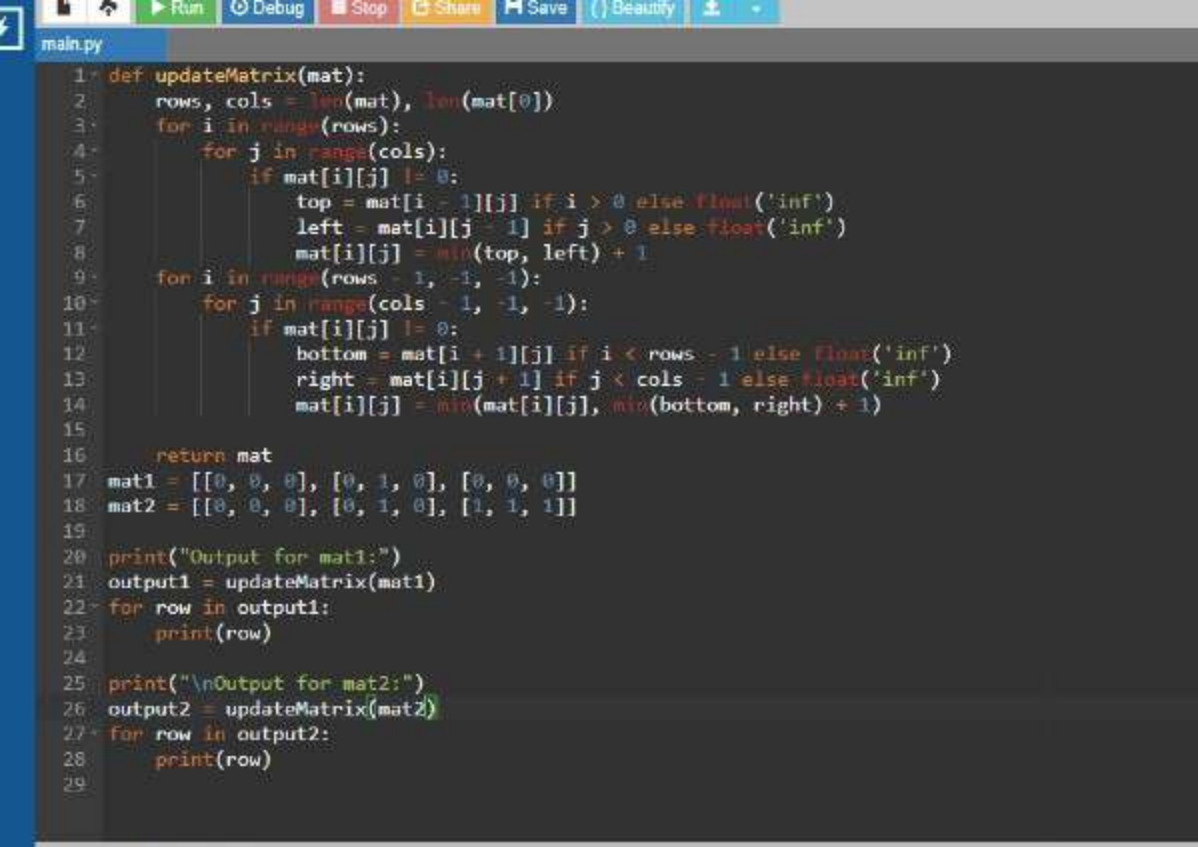
Run | Debug | Stop | Share | Save | {} Beautify

main.py

```python
from bisect import bisect_right
from functools import lru_cache

def min_operations_to_make_increasing(arr1, arr2):
    arr2.sort()

    @lru_cache(None)
    def dp(i, prev):
        if i == len(arr1):
            return 0

        operations = float('inf')

        if arr1[i] > prev:
            operations = dp(i + 1, arr1[i])

        j = bisect_right(arr2, prev)
        if j < len(arr2):
            operations = min(operations, 1 + dp(i + 1, arr2[j]))

        return operations

    result = dp(0, -float('inf'))
    return result if result < float('inf') else -1

arr1 = [1, 5, 3, 6, 7]
arr2 = [1, 3, 2, 4]

print(min_operations_to_make_increasing(arr1, arr2))
```

```
1


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


Run
Debug
Stop
Share
Save
{} Beautify
main.py
def updateMatrix(mat):
rows, cols = len(mat), len(mat[0])
for i in range(rows):
for j in range(cols):
if mat[i][j] != 0:
top = mat[i - 1][j] if i > 0 else float('inf')
left = mat[i][j - 1] if j > 0 else float('inf')
mat[i][j] = min(top, left) + 1
for i in range(rows - 1, -1, -1):
for j in range(cols - 1, -1, -1):
if mat[i][j] != 0:
bottom = mat[i + 1][j] if i < rows - 1 else float('inf')
right = mat[i][j + 1] if j < cols - 1 else float('inf')
mat[i][j] = min(mat[i][j], min(bottom, right) + 1)
return mat
mat1 = [[0, 0, 0], [0, 1, 0], [0, 0, 0]]
mat2 = [[0, 0, 0], [0, 1, 0], [1, 1, 1]]
print("Output for mat1:")
output1 = updateMatrix(mat1)
for row in output1:
print(row)
print("\nOutput for mat2:")
output2 = updateMatrix(mat2)
for row in output2:
print(row)
Output for mat1:
[0, 0, 0]
[0, 1, 0]
[0, 0, 0]
Output for mat2:
[0, 0, 0]
[0, 1, 0]
[1, 2, 1]
...Program finished with exit code 0

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

main.py

```python
def min_length(nums):
    return len(nums) % 2

# Input
nums = list(map(int, input().split()))
print(min_length(nums))

```

input

```
1 2 3 4
0


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

main.py

```python
def string_substrings(words):
    return [w for w in words if any(w in x for x in words if w != x)]

# Input
words = input().split()
print(string_substrings(words))

```

input

```
mass as hero superhero
['as', 'hero']


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

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

main.py

```python
def min_repeats(a, b):
    if b in a:
        return 1
    for i in range(1, len(b)):
        if b.startswith(a[i:]):
            return 1 + i
    return -1

# Example
a = "abcd"
b = "cdabcdab"
output = min_repeats(a, b)
print(output)

```

input

```
3


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