COMP 125: Programming with Python, Fall 2021

Final – January 15, 2022

Question 4 (20 points): 20:25-20:50 (should be submitted by 20:55)

For this question download the file 'Q4.py'. Make sure that the file is located in your current working directory. Implement the function **custom_mask**. This function takes an ndarray (**arr**) and an integer (**n**) as parameters. It returns an ndarray to be used as a custom mask. The mask array has the same dimensions as **arr**. Mask elements are **True**, if their row (i) and column (j) indices satisfy the equation (i + j) % n == 0, and **False** otherwise.

Implement your code in file Q4.py The contents of this file are as follows:

```
import numpy as np

def custom_mask(arr,n):
    '''This function takes an ndarray (arr) and an integer (n)
    as parameters. It returns an ndarray to be used as a custom mask.
    The mask array has the same dimensions as arr.
    Mask elements are True, if their row (i) and column (j) indices
    satisfy the equation (i + j) % n == 0, and False otherwise.'''
    # DO NOT CHANGE THE CODE ABOVE THIS LINE

    # IMPLEMENT THIS FUNCTION

    return

# DO NOT CHANGE THE CODE BELOW THIS LINE
if __name__ == '__main__':
    arr = np.ones((4,6))
    print('Original array')
    print(arr)

    n = 5
    mask = custom_mask(arr,n)
    print('\nCustom mask')
    print(mask)

    arr[mask] = 0
    print('\nUpdated array')
    print(arr)
```

Expected output from the code in Q4.py after implementation: For demo the input array is a 4x6 array of ones, and n=5. Your code should be general, it should work for any array and **n** value.

```
Original array
[[1. 1. 1. 1. 1. 1.]
 [1. 1. 1. 1. 1. 1.]
 [1. 1. 1. 1. 1. 1.]
 [1. 1. 1. 1. 1. 1.]]

Custom mask
[[ True False False False False  True]
 [False False False False  True False]
 [False False False  True False False]
 [False False  True False False False]]

Updated array
[[0. 1. 1. 1. 1. 0.]
 [1. 1. 1. 1. 0. 1.]
 [1. 1. 1. 0. 1. 1.]
 [1. 1. 0. 1. 1. 1.]]
```

For this question, download the Q4.py file from Blackboard. Implement your code in this py file and then upload it to Blackboard before 20:55.