Part 1. Arrays and shapes

Let $A$ be a numpy array of shape $(3,7)$. What is the shape of $a[2:, 1:].transpose()$?

- (A) $(6,2)$
- (B) $(2,6)$
- (C) $(5,3)$
- (D) $(6,3)$
- (E) $(6,)$

Part 2. Row sums

You are given a 2D numpy array in the variable $a$. Produce a version of $a$ that has each row multiplied by a number so that the sum over the row is one. Store it in the variable $a_{\text{normalized}}$.

Use the function numpy.sum to compute the row sums. Its documentation is reproduced (in part) below.

```python
import numpy as np

numpy.sum(a, axis=None)
```

Sum of array elements over a given axis.

Parameters:

- **a**: array. Elements to sum.
- **axis**: None or int. Axis or axes along which a sum is performed. The default (axis = None) is perform a sum over all the dimensions of the input array. axis may be negative, in which case it counts from the last to the first axis.

```python
import numpy as np
```
Part 3. Numpy indexing

Write a piece of code that produces a $10 \times 10$ multiplication table in the variable `mult_table`:

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

Do not use any for loops.

```python
import numpy as np
```