PART - A

My solution:

1. To fine the total number of combinations when rolling two six-sided dice can be calculated by multiplying the number of faces on each die.
2. To represent all possible combinations, I used a nested loop to iterate through each face on Die A and Die B. The value at position (i, j) in the matrix represents the sum of Die A face i and Die B face j.
3. To calculate the probability of each sum, I divided the count of each combination by the total number of combinations. This resulted in a probability matrix.

PART - B

Conditions:

Loki imposed conditions on the dice:

* Die A cannot have more than 4 spots on a face.
* Die A may have multiple faces with the same number of spots.
* Die B can have as many spots on a face as necessary, even more than 6.

My solution:

1. The objective is to reattach the spots to Die A and Die B in a way that the probabilities of obtaining different sums remain the same. We can take probability matrix from Part A as a reference for maintaining probabilities. Each element in this matrix represents the probability of obtaining a specific sum.
2. A transform function ‘undoom\_dice’ is introduced to take Die A and Die B as input and output new faces for both that meet Loki's conditions while preserving the original probabilities.
3. In the ‘generate\_new\_faces’ function, we calculate the expected values for new faces of Die A using the original probability matrix. The rounding and limiting to 4 spots ensure Loki's conditions are met and the output is printed accordingly.