# Word2Vec Task:

For embedding dimension = 10:

Accuracy = 48.75%

For embedding dimension = 100:

Accuracy = 68.75%

For embedding dimension = 1000:

Accuracy = 70%

Conclusion:

I will not run it 5 times for each dimensionality since it takes a long time to do it, but we can observe that the higher the dimensionality the higher the accuracy in this case. I believe it is because since there are many words, if we have a small dimensionality we will not be able to properly represent them all with only that few dimensions, whereas if we have more dimensions it is much easier to find a representation more or less unique for each word. However a problem of higher dimensionality is that it escalates the computational complexity quite rapidly, making it much slower to execute when you have to compare more dimensions in the embeddings.

# Random Indexing With Permutations Task:

For embedding dimension = 1000:

Accuracy = 65%

For embedding dimension = 4000:

Accuracy = 71.25%

For embedding dimension = 10000:

Accuracy = 78.75%

Conclusion:

Same here as we saw above, accuracy increases as so does the dimensionality.

**Something interesting about the window size:** I find it interesting that I have increased the size of the window to 4, and that reduced the accuracy from 70% to 61.25, I had reasons in favor of both it reducing and also it decreasing. I thought it could decrease as with size = 5 that is exactly the structure that we will find in our test data from TOEFL, but also I thought the algorithm would catch more context with a bigger window leading to more accuracy. In any case, I have been able to solve the problem about the fixed size of 2 for the window and fixed bugs when it comes to indexing the lines that are before or after the main line where the word we are evaluating is.