COMP103 Assignment 5 Gene Search

Name: Matthew Corfiatis
Username: CorfiaMatt
ID: 300447277

Core B:
I am using the ArrayList.Get method to get the data from the lists, this has a time complexity of O(1)

Assuming worst case where all the data matches the pattern exactly, e.g
Data: AAAAAAAAAAAAAAAA
Pattern: A

The time complexity would be: O(nm)

Completion B:

If the worst case is still assumed, e.g
Data: AAAAAAAAAAAAAAAA
Pattern: A

The time complexity would be the same O(nm)

Challenge:
How much could this speed up the search?
It would be much slower if you only wanted to do one search because you would search for all 1024 possible subsequences. This method would make searches very very fast O(1) for map lookup, but the generation step at the start is slow. Since the data does not change, the map could be saved in a file so it doe not have to be re-generated each time the data is loaded.

Is it worth it?
I dont think it is worth it because this method effectively searches 1024 times for generation, when you may only want to

search for a few subsequences. I dont think all of the 1024 subsequences would be used. Another reason I think it is not worth it is because you could just save the locations of the subsequences you find so you can look back later instead of running another search.

How much memory space would the map use?
The map would need to store however many matches were found for all of the combinations of the subsequences. The smaller the patter (larger than 0) and the larger the data, the higher the chance for a match so there would be more matches stored in the map.

Would it use more or less memory space if the Map was indexed on all possible subsequences of 8 bases?
Initially I thought it would use more memory. All subsequences using 5 bases has 4^5 = 1024 possibilities, all subsequences using 8 bases has 4^8 = 65536 possibilities. I thought the hash map would have to store all those keys but assuming it just hashes them when they are looked up, I think it would use less space because they aren't stored. I think it would also use less space because there are less match locations at a subsequence length of 8 compared to a subsequence length of 5 that need to be stored

Would this algorithm be helpful for the approximate search? why or why not?
This algorithm would not work for approximate search because a map does not return a value for an approximate key.