The complexity of an algorithm

1. Explain and justify the computational complexity of the algorithm for different values of m and n

A screenshot of a computer

Description automatically generated with medium confidence

* The Big O ( (n\*(n+1)/2) + (m-n)\*n ) only correct with m>=n: The first part which is n\*(n+1)/2) will get the result like a triangle (for example in the image below n is 4)

Letter, calendar

Description automatically generated with medium confidence

However with the code above (example n = 4, and m = 6) it will give the result of

A piece of paper with writing on it

Description automatically generated with medium confidence

And the part that I border is another calculation we need to find, and I see that in each iteration it will have 2 extra elements which is the result of (m-n) and it will iterate 4 times by the loop. Therefore that calculation I have found for that part is (m-n)\*n and plus with the above I have th complexity is Big O ( (n\*(n+1)/2) + (m-n)\*n ) only correct with m>=n.