Course: 76558: Algorithms in Computational Biology
Students: Inbar Neuwirth (214775264), Yishai Ben Zvi (208307843)
Exercise: 1

Part 1: theoretical lower bound on number of possible alignments:

CBIO

1. טענה: יהיו 2 רצפים s,t באורך n. אזי, מס' הסידורים האפשריים הוא לפחות אקספוננציאלי ב-n.

הוכחה: יהיו s,t רצפים באורך n. נבנה את הטבלה הבאה כמו שבנינו בשיעור:

	$-$	s_1	$\cdots$	s_n
$-$	$(0,0)$	$(1,0)$		$(n,0)$
t_1	$(0,1)$			
$\vdots$				
t_n	$(0,n)$			(n,n)

באופן זה מתקיים כי s_i = האות ה-i ברצף s ובאופן דומה מתקיים עבור t. נראה כי התא (i,j) מייצג סידור של הרצפים $s_1,s_2,\dots,s_i$ ו- $t_1,t_2,\dots,t_j$ עבור $i,j\in[0,n]$.

נגדיר 3 צעדים חוקיים:
1. V ($\downarrow$ vertical)
2. H ($\rightarrow$ horizontal)
3. D ($\searrow$ diagonal)

כמו שהגדרנו בכיתה:
נגדיר את P להיות קבוצת כל המסלולים החוקיים.
נגדיר את P' להיות קבוצת כל המסלולים החוקיים המשתמשים רק ב-2 הצעדים V ו-H. וכיוון שכל מסלול גורר לסידור שונה,
$|P|$ = מס' הסידורים הקיימים,
כאשר, מסלול חוקי מוגדר כדלהלן:
1. מתחיל ב-$(0,0)$
2. נגמר ב-(n,n)
3. אחרי תא $(n,*)$ כאשר $n\neq *$, ניתן לבצע צעד H בלבד
4. אחרי תא $(*,n)$ כאשר $n\neq *$ ניתן לבצע צעד V בלבד
5. אחרי תא (n,n) לא ניתן לבצע אף צעד.

אזי מתקיים $P'\subseteq P$, לפי הגדרת הקבוצות, ו- $|P|\geq|P'|$ מ-וודאי.
ומכך נובע כי $|P'|$ הוא חסם תחתון למספר הסידורים הקיימים. נחשב את $|P'|$.

קל לראות כי כדי להגיע מ-$(0,0)$ ל-(n,n) ע"י צעדי V,H בלבד, יש לבצע בדיוק n צעדי V ו-n צעדי H. לכן, גודל הקבוצה P' שווה למספר הקומבינציות של בחירת n מקומות (= של צעדי H או V) מבין $2n$ מקומות. כלומר:

$$|P|\geq|P'|=\binom{2n}{n}=\frac{(2n)!}{(n!)^2}$$

נשתמש בקירוב סטירלינג הידוע כי $n!\approx\sqrt{2\pi n}\left(\frac{n}{e}\right)^n$ ונקבל:

$$\text{מס' הסידורים הקיימים}=|P|\geq|P'|=\frac{(2n)!}{(n!)^2}=\frac{\sqrt{2\pi(2n)}\cdot\left(\frac{2n}{e}\right)^{2n}}{\sqrt{2\pi n}^{\,2}\cdot\left(\left(\frac{n}{e}\right)^{n}\right)^2}$$

$$=\frac{\cancel{2}\sqrt{\pi n}\cdot 2^{2n}\cdot\cancel{\left(\frac{n}{e}\right)^{2n}}}{\cancel{2}\pi n\cdot\cancel{\left(\frac{n}{e}\right)^{2n}}}=\frac{2^{2n}}{\sqrt{\pi n}}=\frac{4^n}{\sqrt{\pi n}}$$

ומצאנו כי $\frac{4^n}{\sqrt{\pi\cdot n}}$ הוא אקספוננציאלי ב-n ולכן כך
שהטענה נכונה, כנדרש.

Part 2: segmentation program:

Complexity Analysis of Our Algorithm

Setup:
The setup phase has a time complexity of **O(n)**. In this step, we precompute the cumulative sums and the sum of squares for the input data. This precomputation allows us to later find the cost of any given segment in constant time, **O(1)**. (see below):

Loop Analysis:
Our algorithm consists of nested loops:

- The **outer loop** runs **O(n)** times.
- Within each iteration of the outer loop, there is an **inner loop** that runs **O(q)** times.
- The operations inside the inner loop are completed in constant time, **O(1)**.

Extract Segments:
This final step has a time complexity of **O(k)**, where k is the number of segments and is less than or equal to n (k ≤ n).

Overall Complexity

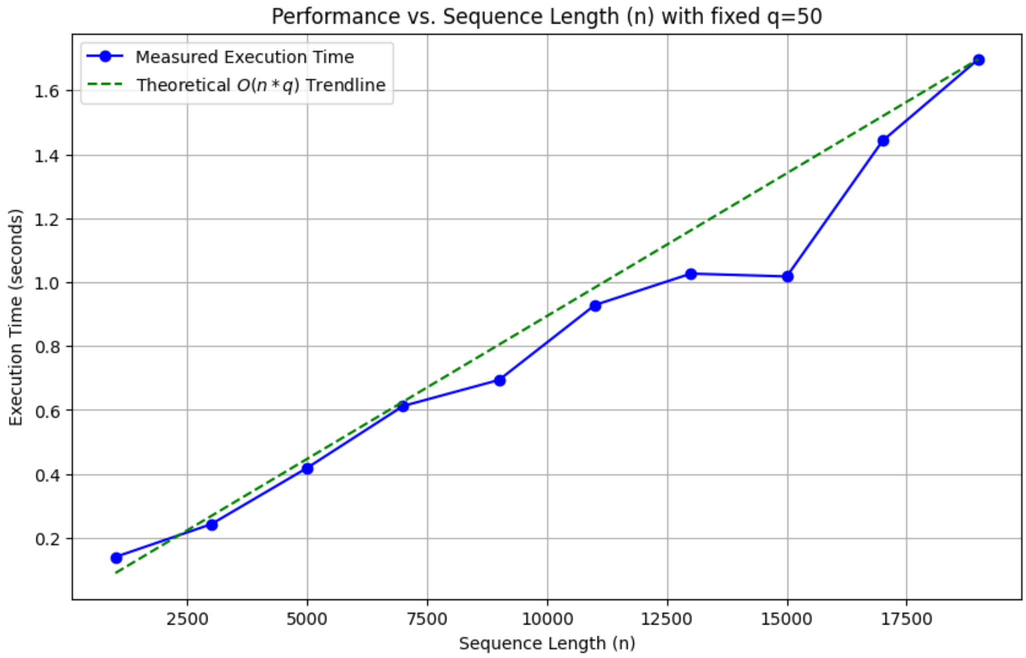
Considering the nested loop structure as the dominant part of the algorithm, the overall time complexity is **O(n * q).**

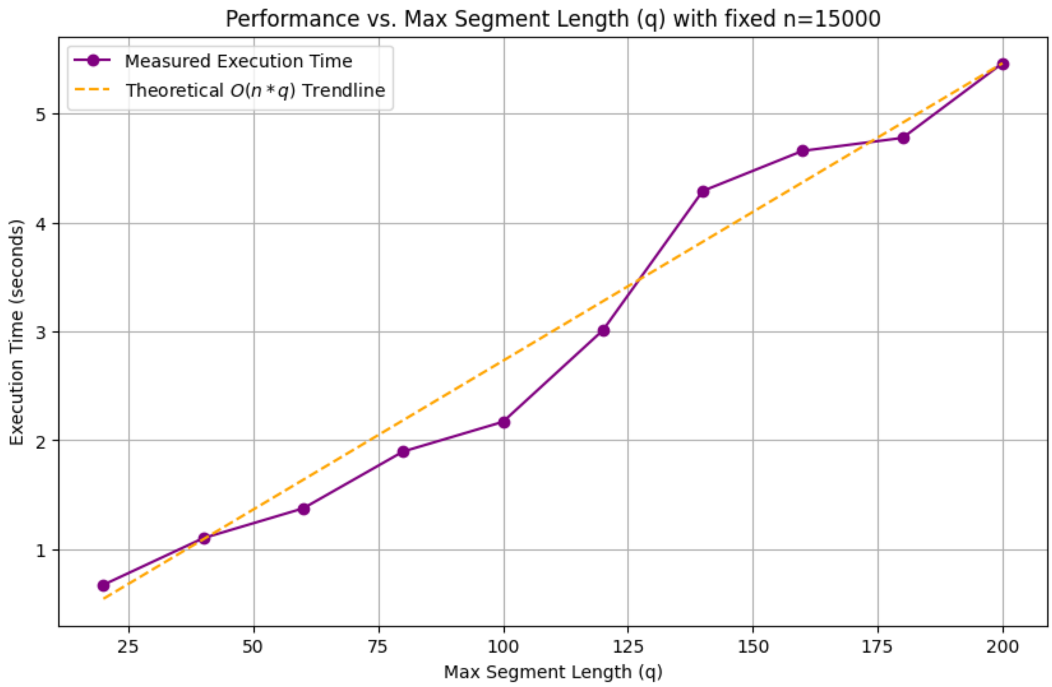
Reference code for efficient segment cost:

```
s = np.zeros(n + 1)
s2 = np.zeros(n + 1)
np.cumsum(x, out=s[1:])
np.cumsum(x**2, out=s2[1:])
def quick_seg_cost(j, i):
    length = i - j
    # sum of values in x[j:i]
    sum_val = s[i] - s[j]
    # sum of squared values in x[j:i]
```

```
sum_sq_val = s2[i] - s2[j]
# Cost = sum(x^2) - (sum(x))^2 / N
return sum_sq_val - (sum_val**2) / length
```

Complexity Visualized: for reference see the plot_complexity function, and plots below:





Appendix: AI usage:

After developing a DP program based on the recursive problem:

$$DP[i] = \min_{\max(0,i-q) < j \le i} \{ DP[j] + \lambda + SSE(j+1, i) \}$$

We used google gemini 2.5 via google ai studio to generate code for the pseudo code structure.