Generating a gene sequence

1. Problem Overview

In bioinformatics and AI, generating a gene sequence that closely matches a desired target sequence (representing the optimal or healthy gene) is a key problem.
In this task, two AI agents take turns selecting nucleotides (gene elements) from a shared pool. Goal of this game is to build a gene sequence that matches a target gene sequence as closely as possible.

2. Game Setup

- Pool: A list of gene elements (e.g, nucleotides like A, T, C, G)
- Target gene: A given target sequence (e.g, "ATCG")
- SID: Student ID, which determines position weight ranges.

3. Turns

- Two agents take turns selecting nucleotides to build the gene sequence. Agent 1 takes the first turn.
- Agent 1 tries to maximize similarity to the target gene.
- Agent 2 tries to minimize that similarity.
- After an agent selects a nucleotide, that nucleotide is removed from the pool, and the next agent must choose from the **remaining nucleotides** in the pool.
- This process continues until the pool is empty.

4. Utility Function

$$\text{Utility} = -\sum_{i=0}^{N-1} \left(w_i \cdot |ASCII(gene[i]) - ASCII(target[i])| \right)$$

- N = Max length of the gene sequence or target; Max(len(gene), len(target))
- W_i = Weight at position i if available, otherwise 1
- ASCII(gene[i]) = ASCII of gene sequence character at i (or 0 if out of bounds)
- ASCII(target[i]) = ASCII of target sequence character at i (or 0 if out of bounds)

5. Weights

- Weights for each position correspond to the **last n digits of your student ID**, where **n is the length of the target.**

Task I

Implement **minimax search with alpha-beta pruning** to generate the best possible gene sequence, given:
- A pool of nucleotides
- A target gene sequence
- Student ID

Input

- The first line contains comma-separated uppercase letters — the initial **nucleotides pool**.
- The second line contains a string — the **target** gene sequence. A target can contain only the elements from the pool. The length of the target should not be greater than your student id.
- The third line contains a space-separated integer— **Your student ID.**

Output

- The first line contains a string — the gene sequence built by the minimax algorithm that gives the best utility score.
- The second line contains a number — the final utility value.

Sample Input and Output

Sample Input	Output
A,T,C,G ATGC 1 8 1 0 4 0 5 2	Best gene sequence generated: AGCT Utility score: -54
A,T,C,G GCAT 2 3 1 8 8 8 1 1	Best gene sequence generated: TGAC Utility score: -153
A,T,C,G CGAT 1 5 0 7 2 2 7 1	Best gene sequence generated: GTAC Utility score: -51

Utility Calculation Breakdown For Task I

Case 1 for sample input 1:
Gene sequence = "ATGC", Target = "ATGC", Weights = [4, 0, 5, 2];
ASCII Values: A=65, T=84, C=67, G=71.

Score $= -(4 \times |A - A| + 0 \times |T - T| + 5 \times |G - G| + 2 \times |G - G|)$
$= -(4 \times |65 - 65| + 0 \times |84 - 84| + 5 \times |71 - 71| + 2 \times |67 - 67|)$
$= -(4 \times 0 + 0 \times 0 + 5 \times 0 + 2 \times 0)$
$= -(0 + 0 + 0 + 0)$
$= 0$

Case 2 for sample input 1:
Gene sequence = ATCG, Target = ATGC, Weights = [4, 0, 5, 2];
ASCII Values: A=65, T=84, C=67, G=71.

Score $= -(4 \times |A - A| + 0 \times |T - T| + 5 \times |C - G| + 2 \times |G - C|)$
$= -(4 \times |65 - 65| + 0 \times |84 - 84| + 5 \times |67 - 71| + 2 \times |71 - 67|)$
$= -(4 \times 0 + 0 \times 0 + 5 \times 4 + 2 \times 4)$
$= -(0 + 0 + 20 + 8)$
$= -28$

Case 3 for sample input 1:
Gene sequence = AGCT, Target = ATGC , Weights = [4, 0, 5, 2]
ASCII Values: A = 65, G = 71, C = 67, T = 84

Score $= -(4 \times |A - A| + 0 \times |G - T| + 5 \times |C - G| + 2 \times |T - C|)$
$= -(4 \times |65 - 65| + 0 \times |71 - 84| + 5 \times |67 - 71| + 2 \times |84 - 67|)$
$= -(4 \times 0 + 0 \times 13 + 5 \times 4 + 2 \times 17)$
$= -(0 + 0 + 20 + 34)$
$= -54$

Case 4:
Gene sequence = AGCTS, Target = ATGC, Weights = [4, 0, 5, 2]
ASCII Values: A = 65, G = 71, C = 67, T = 84, S = 83

Score $= -(4 \times |A - A| + 0 \times |G - T| + 5 \times |C - G| + 2 \times |T - C|$ + **default_weight** $* |S - 0|)$
$= -(4 \times |65 - 65| + 0 \times |71 - 84| + 5 \times |67 - 71| + 2 \times |84 - 67| +$ **1** $\times |83 - 0|)$
$= -(4 \times 0 + 0 \times 13 + 5 \times 4 + 2 \times 17 + 83)$
$= -(0 + 0 + 20 + 34 + 83)$
$= - -137$

*Target[4] = 0 as the gene sequence contains more nucleotides than the target.

Task II

In this part of the game, the pool may include a special nucleotide 'S' at the rightmost position—representing a rare, powerful gene found in golden blood.

Rules:

- If **Agent 1 (Maximizer)** picks 'S', it activates a **genetic booster:**
  - **Genetic Booster:** All **remaining weights** in the evaluation function starting from 'S' position will be multiplied by this equation => (**first 2 digits of your student id / 100)**

- Agent 2 plays normally. Meaning, even if agent 2 encounters nucleotide 'S', it will play normally as it was playing in the previous task (minimize that similarity).

Your goal is to help Agent 1 to decide whether using 'S' leads to a better utility score.

Sample Input and Output

Sample Input	Output
A,T,C,G ATGC 1 8 1 0 4 0 5 2	No With special nucleotide Best gene sequence generated: SCTAG, Utility score: -96.38
A,T,C,G GCAT 2 3 1 8 8 8 1 1	YES With special nucleotide Best gene sequence generated: STGAC, Utility score: -126.11
A,T,C,G CGAT 1 5 0 7 2 2 7 1	NO With special nucleotide Best gene sequence generated: SACGT, Utility score: -94.65

Explanation:
1. NO, because without special nucleotide Score: -54 but With special nucleotide Score: -96.38
2. YES, because without special nucleotide Score: -153 but With special nucleotide Score: -126.11

Utility Calculation Breakdown For Task II

Gene sequence = SCTAG, Target = ATGC, Weights = [4, 0, 5, 2]
Here, S is at position 0. So new weights = [0.72 , 0, 0.9, 0.36] [All weights starting from position 0 are multiplied by the first 2 digits of SID / 100]

ASCII Values: A = 65, G = 71, C = 67, T = 84, S = 83
Score = − (0.72 × |S − A| + 0 ×|C − T| + 0.9 × |T − G| + 0.36 × |A − C| + default_weight × |G - 0|)
      = − (0.72 × |83 − 65| + 0 × |67 − 84| + 0.9 × |84 − 71| + 0.36 × |65 − 67| + 1 × |71 - 0|)
      = − (0.72 × 18 + 0 × 17 + 0.9 × 13 + 0.36 × 2 + 1 × 71)
      = − (12.96 + 0 + 11.7 + 0.72 + 71)
      = −96.38

**If gene sequence is CTSAG, weights will be [4, 0, 0.9, 0.36]. (All weights starting from position 2 are multiplied by [first 2 digits of SID / 100])