

Relatório 1º projeto ASA 2025/2026

Grupo: AL065

Aluno(s): Gonalo Rafael (113528) e Nuno Teixeira (114559)

Descrio do Problema e da Soluo

A soluo apresentada trata-se de um caso de programaco dinmica com recurso a tabulao (bottom-up), onde se optou por descobrir o ltimo aminocido a ser removido para encontrar a energia mxima, visto que comeando pelo primeiro criaria dependncias que causariam um problema na resoluo, pois os subproblemas $[l, k-1]$ e $[k+1, r]$ teriam o aminocido em $k-1$ e $k+1$ como vizinhos. Por isso, resolveu-se escolher o ltimo aminocido a ser removido em cada um dos subproblemas, tornando-os independentes.

Comea-se por receber os dados do input onde a sequncia de aminocidos  convertida para o ndice correspondente na tabela de afinidades de aminocidos. De seguida, criam-se duas matrizes, *optimal* e *cache*, de forma a guardar tanto a energia mxima para um determinado intervalo, tal como o ndice do aminocido timo. Definem-se as variveis temporrias *total*, *maxEnergia* e *maxK* de modo a tornar o cdigo mais eficiente (o que evita estar constantemente a encontrar o endereo nas matrizes *optimal* e *cache*), entra-se, por fim, no *loop*. Dentro dos *fors* atribui-se s respetivas tabelas os valores de energia tima de remoo e do ndice do aminocido timo a remover. Ao completar as tabelas, faz-se um ltimo passo para conseguir a sequncia tima de remoo, utilizando uma recurso. De modo a evitar problemas com nmeros muito grandes (e sabendo que os nmeros no sero negativos) utiliza-se unsigned long long para que no ocorra nenhum problema de overflow.

Anlise Terica

Leitura de dados: $O(n)$

Leitura input potenciais	$O(n)$
Leitura input sequncia + converso:	$O(n)$

Processamento das instncias: $O(n^2)$

Init cache[n+2][n+2]	$O(n^2)$
Init optimal[n+2][n+2]	$O(n^2)$

Algoritmo: $O(n^3)$

For l = n to 1	$O(n)$
For r = 1 to n	$O(n)$
maxEnergia = 0	$O(1)$
maxK = 1	$O(1)$
For k = 1 to r	$O(n)$
total = Custo Remoo	$O(1)$
if(total > maxEnergia)	$O(1)$
maxEnergia = total	$O(1)$
maxK = k	$O(1)$
if(total == maxEnergia && k >= maxK)	$O(1)$
maxK = K	$O(1)$
cache[l][r] = maxEnergia	$O(1)$
optimal[l][r] = k	$O(1)$

Relatório 1º projeto ASA 2025/2026

Grupo: AL065

Aluno(s): Gonalo Rafael (113528) e Nuno Teixeira (114559)

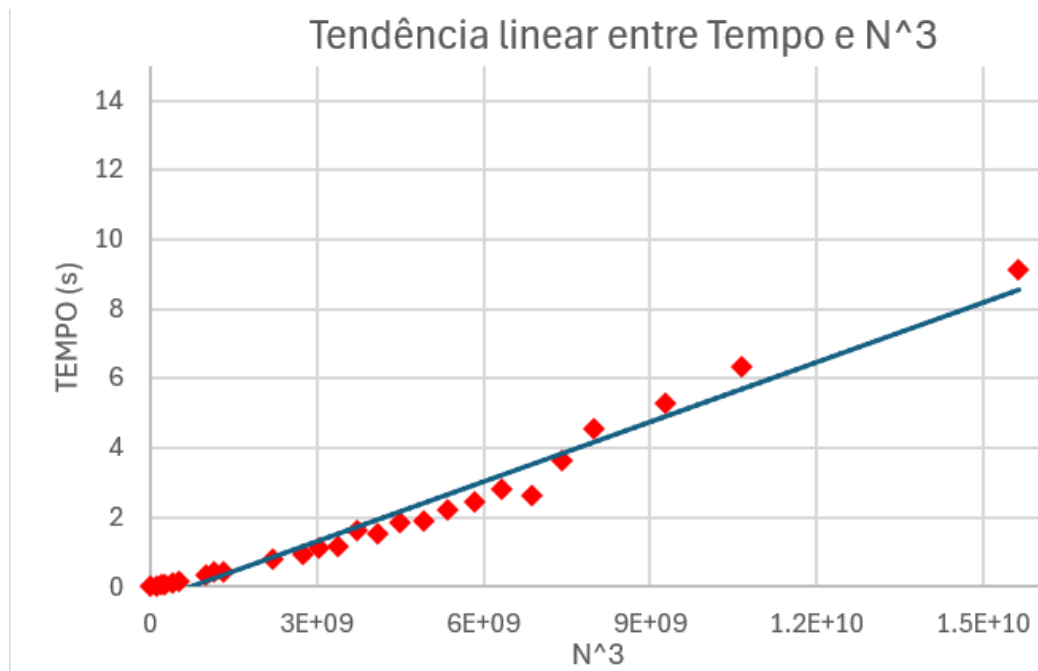
Output: $O(n)$

printEnergia	$O(1)$
Computar Sequência Ótima	$O(n)$

Complexidade global: $O(n^3)$

Avaliação Experimental dos Resultados

Foram geradas 26 instâncias de tamanho incremental, de $n=100$ até $n=2500$. Observáveis no gráfico, onde o eixo dos XX representa o n^3 , e o eixo dos YY o tempo.



É possível observar uma relação linear, que confirma a complexidade global $O(n^3)$.