Permutações e n-rainhas

Carlos Renato Medeiros*
Depto de Matemática, FEIS, UNESP
15385-000, Ilha Solteira, SP
e-mail: c4rl4o@gmail.com

Edson Donizete de Carvalho
Universidade Estadual Paulista - Departamento de Matemática
15385-000, Campus de Ilha Solteira, Ilha Solteira, SP
e-mail: edson@mat.feis.unesp.br

RESUMO

Problemas computacionais, em geral, possuem riquezas em suas soluções como na variedade de soluções, tipo de algoritmo (iterativo, recursivo, ...), tempo e memória gastos e associação com outros problemas. Este é o caso dos problemas clássicos das n-rainhas e das permutações, com várias soluções propostas. Neste trabalho, propomos uma solução para as permutações vinculado às n-rainhas e uma variação desta.

No problema das permutações, desejamos obter todas as permutações dos caracteres de uma cadeia de caracteres sem repetição de tamanho n. Sabemos da Análise Combinatória que tal cadeia possui $n!$ permutações. Consideraremos o caso em que a cadeia é da forma $(1, 2, ..., n)$.

O problema das n-rainhas consiste em dispor n rainhas do jogo de xadrez em um tabuleiro $n \times n$, de forma que nenhuma rainha ataque a outra conforme as regras do jogo, isto é, não teremos duas rainhas ocupando uma mesma linha, coluna e ou diagonal. O caso $n = 1$ é trivial e os casos $n = 2$ e $n = 3$ não possuem solução, então consideraremos $n \geq 4$. Como temos uma rainha por linha e coluna, indicaremos pelo vetor $V = (v_1, v_2, ..., v_n)$ as posições das n-rainhas no tabuleiro $n \times n$, sendo que cada v_i assume valores no conjunto $\{1, 2, ..., n\}$ e $v_i \neq v_j$, $i \neq j$. Esta representação é única, sendo que o valor $V[i] = v_i = j$ representa a coluna j em que a rainha da linha i está. Também é possível obter uma representação matricial M do vetor V fazendo $M[i, V[i]] = 1$ e $M[l, c] = 0$ nos demais casos. Esta representação matricial permite uma interpretação direta das posições das rainhas no tabuleiro, interpretando o número 1 como sendo uma rainha e o número 0 como uma casa vazia.

Da representação vetorial V, segue que não teremos duas rainhas em uma mesma linha ou coluna, logo, basta verificarmos as diagonais, ou seja, devemos ter $|i - j| \neq |V[i] - V[j]|$, para todo $i \neq j$. Uma solução natural seria gerar todas as cadeias $(v_1, v_2, ..., v_n) \subset P_n$ e verificar cada uma, sendo P_n o conjunto de todas as permutações de $(1, 2, ..., n)$. Assim, estabelecemos um vínculo de dependência das n-rainhas em função das permutações.

Uma forma de resolver o problema das n-rainhas é através de *backtracking*. Em um momento, todas as rainhas $(v_1, v_2, ..., v_{i-1})$ estão posicionadas e queremos posicionar a rainha da linha i, então fazemos $V[i] := 1$, caso esta rainha esteja na mesma coluna ou diagonal de uma outra, então fazemos $V[i] := V[i] + 1$ até que não haja mais ataque e prosseguimos para a rainha da linha $i+1$. Caso a rainha da linha i atinja a coluna n e ainda exista ataque, então retornamos para a rainha da linha $i - 1$ e repetimos o processo para esta. O algoritmo termina quando a rainha da primeira linha atinge a coluna $n + 1$.

Neste algoritmo, se compararmos apenas as colunas, então teremos um problema de n-torres e todas as soluções deste problema são todas as permutações da cadeia $(1, 2, ..., n)$, assim, estabelecemos outro vínculo, desta vez, das permutações em função das n-torres, logo, a solução para o problema das permutações é solução do problema das n-torres, por sua vez, um caso geral das n-rainhas.

Referências

CORMEN, T. H., et al. *Algoritmos - teoria e prática.* Tradução da 2ª edição americana. CAMPUS, 2002

*bolsista de Iniciação Científica FAPESP