UNIVERSIDADE ESTADUAL DA PARAÍBA

CAMPINA GRANDE, 30 DE MAIO DE 2022

DEPARTAMENTO DE COMPUTAÇÃO - CCT

DISCIPLINA - TÉCNICAS DE ANÁLISE DE ALGORITMO

PROFESSOR - PAULO CÉSAR OLIVEIRA BRITO

**1 -** Como o backtracking consegue efetuar uma busca eficiente, apesar de não ser exaustiva?

* Como o backtracking realiza suas buscas com um direcionamento a chegar em seu objetivo, ou seja, no sucesso, acaba por excluir muitas buscas sem que elas precisem ser examinadas de forma explícita.

**2 -** Explique como backtracking e recursividade são utilizados para resolver o problema do passeio do cavalo.

* O passeio do cavalo trata de um desafio referente a fazer com que a peça do tabuleiro “cavalo” passe por todos os pontos do tabuleiro uma única vez, realizando ao todo movimentos. Logo, a partir do backtracking e da recursividade é possível tentar chegar no resultado esperado a partir da montagem de um tabuleiro (com uma matriz 8x8) e a declaração de uma variável inteira que será incrementada a cada movimento até que o cavalo passe por todos os pontos do tabuleiro. A recursividade entra no momento de realizar os possíveis movimentos do cavalo de forma continua e o backtracking na analise referente ao movimento do cavalo, procurando analisar se esse movimento vai possibilitar com que a solução seja atingida, caso o backtracking mostre que não chegará na solução desejada, novas tentativas serão realizadas.

**3 -** Analise as afirmativas a seguir a respeito de algoritmos recursivos.

I. Diz-se que uma rotina é recursiva se a sua definição envolver uma chamada a ela mesma. Neste sentido, o termo recursão é equivalente ao termo indução utilizado por matemáticos.

II. Cada algoritmo recursivo possui um algoritmo iterativo equivalente e vice-versa, mas que pode ter mais ou menos complexidade em sua construção.

III. Uma função recursiva possui duas partes: caso base e caso recursivo.

IV. Um algoritmo pode ser chamado de iterativo quando ele requer a repetição implícita de um processo até que determinada condição seja satisfeita.

V. A recursividade possibilita a escrita de um código mais enxuto, com maior legibilidade e simplicidade.

Assinale a alternativa que possui alguma afirmação INCORRETA.

1. I e II.
2. I e V.
3. II e III.
4. **III e IV. ✓**

**4 -** Uma boa lógica de programação é fundamental para que os algoritmos sejam bem desenvolvidos e, consequentemente, os programas bem implementados, claro que se agregando o conhecimento da sintaxe da linguagem de programação escolhida. Dessa forma, pensando-se em estruturas de algumas implementações, considere o seguinte trecho de código:

int fibonacci (int N)

{

if (N <= 1)

return N;

else

return ( fibonacci(N - 1) + fibonacci(N - 2) );

}

Pode-se afirmar, a partir do código anteriormente apresentado, que:

1. existe uma estrutura de repetição.
2. existe uma estrutura de desvio múltipla.
3. **existe um processo recursivo. ✓**
4. existe um chamada de função com passagem de parâmetros por referência.