Universidade do Estado do Rio Grande do Norte

Departamento de Informática

Curso de Ciência da Computação

Lista 3 de Atividades da Disciplina de Lógica e Matemática Discreta

Data: 26/06/2024

Prof. Antônio Oliveira Filho

1. Defina o ponto de parada para o algoritmo a^n.
2. Defina a recorrência do algoritmo do algoritmo da potência.
3. Mostre os estados da pilha de execução do algoritmo da potência para a = 2 e n = 3.
4. Defina o ponto de parada para o algoritmo do Fatorial.
5. Defina a recorrência para o algoritmo do Fatorial.
6. Mostre os estados da pilha de execução para o algoritmo do Fatorial para n = 3.
7. Defina a recorrência para a série Fibonacci.
8. Defina os pontos de parada para o algoritmo Fibonacci.
9. Mostre o estado da pilha de execução para o algoritmo Fibonacci.
10. Defina a recorrência para o n-ésimo termo de uma PA.
11. Defina os pontos de parada para o algoritmo da PA.
12. Mostre o estado da pilha de execução para o algoritmo da PA.
13. Defina a recorrência para o n-ésimo termo de uma PG.
14. Defina os pontos de parada para o algoritmo PG.
15. Mostre o estado da pilha de execução para o algoritmo PG.
16. Defina a condição de parada para a soma dos elementos de um array.
17. Defina a recorrência para a soma dos elementos de um array.
18. Mostre os estados da pilha de execução para a soma dos elementos de um array.
19. Faça um algoritmo para criar uma cópia invertida de um vetor de números inteiros:
    - Entrada: {1,2,3,4,5,6,7}
    - Saída: {7,6,5,4,3,2,1}
20. Faça um algoritmo para contar quantos números pares há num vetor de inteiros.
    - Entrada: {1,2,3,4,5,6,7}
    - Saída: 3
21. Faça um algoritmo que responde “primo” se o número informado for primo ou “não primo”, caso contrário. Um número primo é um inteiro que é divisível apenas por si e por 1.

    - Entrada: 2
    - Saída: primo
    - Entrada: 4
    - Saída: não primo
22. Faça um algoritmo para decompor um número em todos os seus divisores. Armazene o resultado em um array.
    - Entrada: 12
    - Saida: {1,2,3,4,6,12}
23. Faça um algoritmo para iniciar uma matriz com zeros.
    - Entrada: {{null,null,null},{null.null.null}}
    - Saída: {{0,0,0},{0.0.0}}
24. Faça um algoritmo para iniciar uma diagonal principal de uma matriz com 1.

    Entrada:

    {{null,null,null},

    {null,null,null},

    {null,null,null}}

    Saída:

    {{1,null,null},

    {null,1,null},

    {null,null,1}}

25. Faça os estados da pilha do algoritmos recursivo descrito a seguir:

```
caminha(b) {
    if(b.prox==null) {
        return b
    } else {
        caminha(b.prox)
    }
}
```

26. O que se pode afirmar sobre a complexidade de um algoritmo que procura por um dado número em um array:

    - Quando se sabe o índice;
    - Quando não se sabe o índice;
27. O que se pode afirmar sobre a complexidade de um algoritmo que procura por um dado número em uma matriz:
    - Quando se sabe o índice;
    - Quando não se sabe o índice;
28. Sabendo que o array da questão 26 está ordenado de modo crescente, que abordagem simples poderia reduzir a complexidade e qual seria a nova complexidade?
29. Uma matriz esparsa é a que tem muitos espaços não preenchidos. Faça um algoritmo que transforma uma matriz esparsa em um array que contém apenas os valores diferentes de null.
30. O que teria que ser feito na questão 29 para preservar a posição original dos valores.
31. Implemente a versão recursiva do famoso algoritmo de Euclides para o cálculo do Máximo Divisor Comum (MDC) de um número. A versão iterativa dele é:
    - Função MDC(a, b)
    - Enquanto b ≠ 0
    - temp ← b
    - b ← a % b
    - a ← temp
    - Fim Enquanto
    - Retornar a
    - Fim Função
32. Se a equação da forma ax + by = c com a, b e c inteiros é uma equação diofantina linear se e somente se o MDC(a,b) dividir c. Faça um algoritmo que mostre se a equação: 56x + 42y = 14 é diofantina.
33. Explique o se somente se da questão 32.
34. Comente a afirmação: se a e b forem primos entre si a equação será sempre diofantina independentemente do valor inteiro c.
35. Pesquise sobre a complexidade computacional do algoritmo para o cálculo do MDC(a,b) de Euclides.