MINISTÉRIO DA EDUCAÇÃO
Universidade Federal de Ouro Preto
Departamento de Computação e Sistemas
Campus João Monlevade

NOÇÕES DE ANÁLISE DE COMPLEXIDADE DE ALGORITMOS

Curso: Engenharia Elétrica — 1º semestre de 2023
Disciplina: Algoritmos e Estrutura de Dados I — Data: 12/05/23
Professor: Alexandre Magno de Sousa

1 FUNÇÕES DE CUSTO E ANÁLISE ASSINTÓTICA

1. Seja o trecho de código da função a seguir

```c
void BubbleSort(int A[], int n){
 int x;
 int i,j;
 for(i = 1; i < n; i++){
  for(j = n; j > i; j--){
     if(A[j] < A[j - 1]){
        x = A[j];
        A[j] = A[j - 1];
        A[j - 1] = x;
     }
  }
 }
}
```

apresente para essa função a análise de complexidade do custo do número de atribuições para todos os casos (pior, melhor e caso médio).

2. Determine a função do custo do número de atribuições de algoritmos que realizam: (a) a operação de multiplicação entre matrizes; (b) a técnica de eliminação de Gauss para transformar uma matriz em triangular superior; e (c) o cálculo de determinantes. Para isso, construa uma função em Linguagem C que realize cada uma dessas operações. **Observação**: a matriz pode ser de ordem 4 ou superior, porém, sua dimensão será dada por $n \times n$, em que n é a ordem da matriz.

3. Encontre o custo de complexidade da função usada para encontrar o k-ésimo inteiro em um vetor não ordenado de inteiros:

```c
int selec_kesimo (int a[ ], int k, int n){
    int i, j, mini, tmp;
    for (i = 0; i < k; i++){
        mini = i;
        for (j = i + 1; j < n; j++)
            if (a[j] < a[mini])
                mini = j;
        tmp = a[i];
        a[i] = a[mini];
        a[mini] = tmp;
    }
    return a[k - 1];
}
```

apresente para essa função a análise de complexidade do custo do número de atribuições, além disso, para essa função, responda se existem pior caso, melhor caso e caso médio.

4. Para cada função a seguir encontre os pares de valores para c e n_0 para cada função de acordo com o limite assintótico superior e inferior, isto é, O-grande e Ω-grande:
   (a) $f(n) = 3n^3 + 2n^2 + n$.
   (b) $f(n) = 2n^2 + 4n - 15$.
   (c) $f(n) = 3n + 7$.
   (d) $g(n) = 3n^2 + 1$.
   (e) $f(n) = 5 \log_2 n$.

5. Classifique cada par de funções $f(n)$ e $g(n)$ a seguir para verificar se $g(n)$ é um limite assintótico para $f(n)$, tais como $f(n) = O(g(n))$, $f(n) = \Omega(g(n))$ e $f(n) = \Theta(g(n))$:
   (a) $f(n) = 3n + 7$ e $g(n) = 5n + 2$.
   (b) $f(n) = 5 \log_2 n + 7$ e $g(n) = 5n + 1$.
   (c) $f(n) = 5 \times 2^n + 3$ e $g(n) = 3n^2 + 5n$.
   (d) $f(n) = 5n + 7$ e $g(n) = 3n^2 + 1$.
   (e) $f(n) = 5n + 7$ e $g(n) = 3n^2 + 1$.
   (f) $f(n) = n^2 \times 3^n + 7$ e $g(n) = n^3 \times 2^n + 1$.

6. Qual das seguintes afirmações sobre o crescimento assintótico das funções não é verdadeira:
   (a) $2n^2 + 3n + 1 = O(n^2)$.
   (b) $\log n^2 = O(\log n)$.
   (c) Se $f(n) = O(g(n))$ e $g(n) = O(h(n))$, então $f(n) = O(h(n))$.
   (d) Se $f(n) = O(g(n))$, então $g(n) = O(f(n))$
   (e) $2^{n+1} = O(2^n)$.
   (f) $2^{2n} = O(2^n)$.
   (g) $f(n) = O(u(n))$ e $g(n) = O(v(n))$ então $f(n) + g(n) = O(u(n) + v(n))$.
   (h) $f(n) = O(u(n))$ e $g(n) = O(v(n))$ então $f(n) - g(n) = O(u(n) - v(n))$.

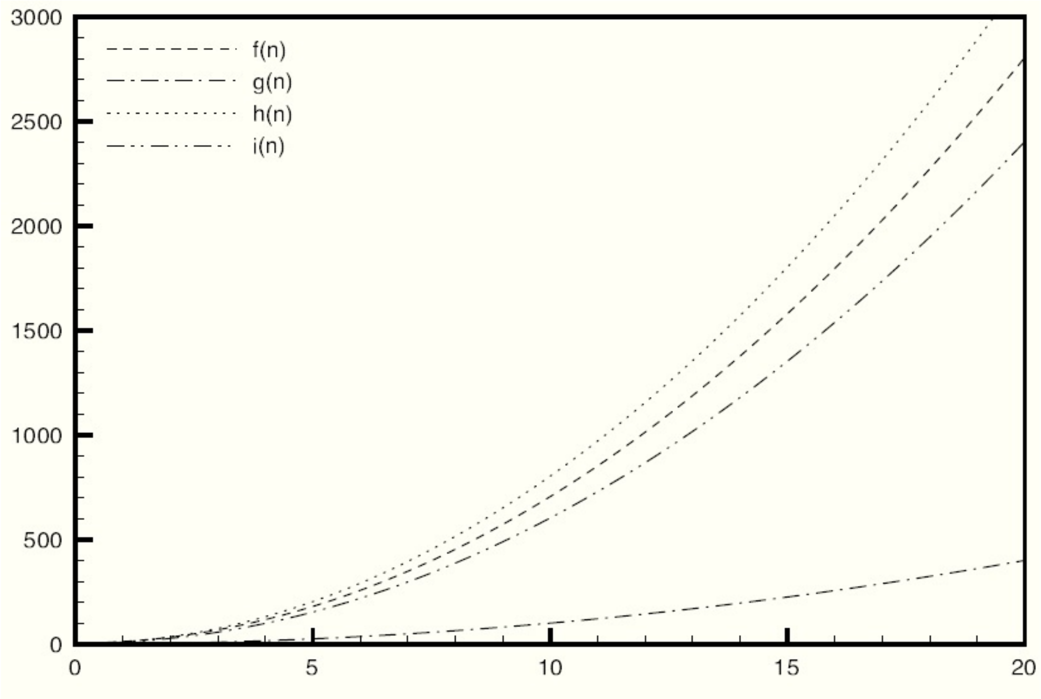

Figura 1: comportamento assintótico das funções $f(n)$, $g(n)$, $h(n)$ e $i(n)$.

7. Observe as funções representadas no gráfico da Figura 1. Assinale a afirmativa FALSA sobre o crescimento assintótico dessas funções:

(a) $f(n) = O(h(n))$ e $i(n) = \Omega(g(n))$.

(b) $f(n) = \Theta(h(n))$ e $i(n) = \Omega(h(n))$.

(c) $g(n) = O(i(n))$ e $h(n) = \Omega(g(n))$.

(d) $g(n) = O(i(n))$, $i(n) = O(f(n))$ e, portanto, $g(n) = O(f(n))$.

(e) $h(n) = \Omega(i(n))$, logo, $i(n) = O(h(n))$.

8. Na tabela a seguir, indique se $f(n)$ é O, o, Ω, ω ou Θ de $g(n)$ para cada par de expressões $f(n)$ e $g(n)$ na tabela a seguir. Considere a constante $c > 1$. Para isso, preencha sua resposta na tabela com Sim ou Não. Apresente uma justificativa para cada linha da tabela.

Tabela 1: Funções $f(n)$ e $g(n)$ e suas relações assintóticas.

$f(n)$	$g(n)$	O	o	Ω	ω	Θ
$\log\log n$	n^c					
2^n	n^n					
n^2	$n\log n$					
$n^{\log m}$	$m^{\log n}$					
$\log n!$	$\log n^n$					

Assuma que $\log n! = \log 1 \times 2 \times 3 \times \ldots \times n = \log 1 + \log 2 + \log 3 + \ldots + \log n \approx n \log n$.