Aluno: Iago Costa dos Flores
Disciplina: Complexidade de Algoritmos
Professor: Manoel Ribeiro
Turma: Engenharia da Computação 2018

Prova 01 19/06/2021

01-

	$200n^2$	22^n
01	200	22
02	800	484
03	1800	10.648
04	3.200	234.256
05	5.000	
06	7200	
07	9.800	

Até $n=2$ algoritmo 22^n é mais eficiente

Apartir de $n=3$ algoritmo $200n^2$ é mais eficiente

02-

$$A = \begin{cases} C = 2n^2 \\ T_a = t \\ n = x \\ V_c = 1 \end{cases} \qquad T_a = 2x^2$$

$$2n^2 = 2x^2$$

$$\boxed{n = x}$$

$$B = \begin{cases} C = 2n^2 \\ T_B = t \\ n = 3x \\ V_c = 30 \end{cases} \qquad T_B = \frac{2\cdot(3x)^2}{30}$$

$$2n^2 = \frac{2\cdot(3x)^2}{30}$$

$$2n^2 = \frac{2\cdot 9x^2}{30}$$

$$n = \frac{9x}{30}$$

$$\boxed{n = 0{,}3x}$$

$$t_a = t_b$$

$$t_a = 2x^2$$

$$t_b = \frac{2\cdot 9x^2}{30}$$

$$t_b = \frac{9\cdot t_a}{30}$$

$$\boxed{t_b = 0{,}3\,t_a}$$

03)

[1, 2, 3, 4, 5] ⟶ [1, 3, 6, 10, 15]

Soma dos anteriores (A, n)

0	for i = 2 to n	C_0	n
1	soma = 0	C_1	n - 1
2	soma = soma + vet[i]	C_2	n - 1
3	soma = soma + vet[i-1]	C_3	n - 1
4	A[i] = soma	C_4	n - 1
5	return A	C_5	1

04)

invariante de laço: o vetor de retorno da função estará sempre atualizado com a soma do índice atual da iteração com os índices anteriores.

inicialização: antes da primeira interação do for, a primeira posição do vetor já vai está atualizada com a soma dele próprio com os índices anteriores que não existem.

manutenção: Após cada iteração todos os índices anteriores até o atual do loop terá a soma dele mesmo mais os valores anteriores.

término: No final todo o vetor estará atualizado com a soma de cada índice mais os seus anteriores.

05)

$$T(n) = nC_0 + (n-1)C_1 + (n-1)C_2 + (n-1)C_3 + (n-1)C_4 + 1C_5$$

$$T(n) = nC_0 + (n-1)(C_1 + C_2 + C_3 + C_4) + C_5 \leftarrow$$ Pior caso percorrer todo o vetor

Melhor caso vetor com posição 1:

$$T(n) = C_0 + C_5$$

Melhor caso vetor com posição 2+:

$$T(n) = nC_0 + (n-1)(C_1 + C_2 + C_3 + C_4) + C_5$$