UNIVERSIDADE FEDERAL DE SANTA CATARINA – UFSC- CTC
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
PROJETO E ANÁLISE DE ALGORITMO
Prof. Alexandre Gonçalves Silva
Aluno: Osmar de Oliveira Braz Junior

Questão 7

7. Suponha que, para entradas de tamanho *n*, você tenha que escolher um dentre três algoritmos A, B e C.
(a) Algoritmo A resolve problemas dividindo-os em cinco subproblemas de metade do tamanho, recursivamente resolve cada subproblema e então combina as soluções em tempo O(n).
(b) Algoritmo B resolve problemas dividindo-os em dois subproblemas de tamanho n - 1, recursivamente resolve cada subproblema e então combina as soluções em tempo O(1).
(c) Algoritmo C resolve problemas dividindo-os em nove subproblemas de tamanho n/3, recursivamente resolve cada subproblema e então combina as soluções em tempo O(n2).

Qual o consumo de tempo de cada um desses algoritmos? Expresse as suas respostas em termos da notação O, mas procure dar as respostas com funções para limites superiores mais próximos possíveis. Qual algoritmo é assintoticamente mais eficiente no pior caso? Justifique as suas respostas.

$\boldsymbol{(a)\ T(n) = 5T\left(\frac{n}{2}\right) + \mathrm{O}(n)}$

R.:
a=5
b=2
k=1 //Expoente de n
f(n) = O(n)
Θ($n^{log_b a}$)= Θ($n^{log_2 5}$) => $log_2 5$ =2,321928095

Comparação #1:
f(n) ∈ O(($n^{log_b a-\varepsilon}$))
n^1 = $n^{log_b a-\varepsilon}$)
$1 = log_b a - \varepsilon$
$b^1 = b^{log_b a-\varepsilon}$
b^1=$a - \varepsilon$
2=$(5 - \varepsilon)$
$\varepsilon = 3$
ε > 0 (Atende o caso 1)

Comparação #2:
f(n) = Θ($n^{log_b a}$) ou seja,
n <> $n^{log_2 5}$ // Não é igual portanto não é o Caso 2 do Teorema Mestre
$log_b a < k$ //Caso 3
$log_2 5 < k$
2,321928095< 1 //Portanto não é o caso 3

$log_b\, a > k$ //Caso 1
$log_2\, 5 > k$
2,321928095> 1 //Portanto é o caso 1

Temos o Caso 1 do Teorema Mestre, portanto
T(n) = 4T(n/2) + O(n)
Então T(n)∈ $\Theta\left(n^{log_b\, a}\right) = \Theta\left(n^{log_2\, 5}\right)$

(b) $\boldsymbol{T(n) = 2T(n-1) + O(1)}$
R.: Não dá para usar o teorema mestre.

Resolvendo através de expansão telescópica. (segundos slides aula0901)
1) Fórmula original
$T(n) = 2T(n-1) + O(1)$

2) T(n) está escrito em função de T(n-1)

3) Isole as equações para T(n-1) e T(n-2):
$T(n-1) = 2T(n-2) + 1$
$T(n-2) = 2T(n-3) + 1$

4) Substitua T(n-1) pelo valor que foi isolado acima e, em seguida, o mesmo para T(n-2)

- Substituindo o valor isolado de T(n-1) em
  $T(n) = 2T(n-2) + 1)$
  Temos
  $T(n) = 2(2T(n-2) + 1) + 1$
  Simplificando
  $T(n) = 2^2T(n-2) + 2 * 1\ + 1$
  $T(n) = 2^2T(n-2) + 2 * 1\ + 1$
  $T(n) = 2^2T(n-2) + 3$

- Agora substituindo o valor de T(n-2) em
  $T(n) = 2(2T(n-2) + 1) + 1$
  Temos
  $T(n) = 2(2(2T(n-3) + 1) + 1) + 1$
  Simplificando
  $T(n) = 2^2(2T(n-3) + 1) + 2 * 1 + 1$
  $T(n) = 2^3T(n-3) + 2^2 * 1 + 2^1 * 1 + 1$
  $T(n) = 2^3T(n-3) + 2^2 * 1 + 2^1 * 1 + 2^0$
  $T(n) = 2^3T(n-3) + [2^2 + 2^1 + 2^0]$

5) Identifique a fórmula do i-ésimo passo

T(n) = $2^iT(n-i) + 2^{i-1}$
T(n) = $2^iT(n-i)$ +Soma da PG a1=1 e q=2]
T(n) = $2^iT(n-i) + [\sum_{i=0}^{(k-1)}(2)^i]$

Calculado para n=4
T(4) = 2T(4-1) +2^3 //i=3
T(4) = 2T(3) + 8

T(3) = 2T(3 - 1) + 2^2 //i=2
T(3) = 2T(2) + 4

T(2) = 2T(2-1) + 2
T(2) = 2T(1) + 2=> T(1) = 1

T(1) = $1 + 1$

T(2) = $2 * 1 + 1 + 2$
T(2) = $2 + 1 + 2$
T(3) = $2T(2) + 4$
T(3) = $2 * \ 2 + 1 + 2 \ + 4$
T(4) = 2T(3) + 8
T(4) = $2 * \ 2 * \ 2 + 1 + 2 + 4 + 8$
T(4) = $2^3 + 15$
T(4) = $2^3 + 15$

6) Descubra o valor de i de forma a igualar o parâmetro de T(x) ao parâmetro (valor de n) no caso base
T(n - i) = T(1) = 1
n - i = 1
-i = -n+1
i =n-1

7) Substitua o valor de i na fórmula do i-ésimo caso
T(n) = $2^i T(n - i) + [2^{i-1} + 2^{i-2} + \cdots + 2^0]$

Forma geral
T(n) = $2^3 T(n - i) + 2^2 + 2^1 + 2^0$

Considerando
$\sum_{i=0}^{k} x^i = 1 + x + x^2 + \ldots + \ x^k$ //A.5 Página 832 CLRS (3ed)
$\sum_{i=0}^{k} x^i = \frac{x^{k+1}-1}{x-1}$
A série geométrica substituir k=k-1
$\sum_{i=0}^{k-1} 2^i = 2^2 + 2^1 + 2^0$
Substituindo na fórmula do somatório
$\sum_{i=0}^{k-1} 2^i = \left(\frac{2^{k-1+1}-1}{2-1}\right)$
$\sum_{i=0}^{k} 2^i = \left(2^k - 1\right)$
Substituindo k = $\mathrm{i} - 1$
$\sum_{i=0}^{k-1} 2^i = \left(2^{i-1+1} - 1\right)$
$\sum_{i=0}^{(k-1)} 2^i = \left(2^i - 1\right)$

//Substituindo o resultado do somatório na fórmula do i-ésimo caso
T(n) = $2^i T(n - i) + \left(2^i - 1\right)$
Considerando i= $n - 1$
T(n) = $2^{n-1} T(1) + 2^{n-1} - 1$ //T(1) =1 por omissão
T(n) = $2^{n-1} + 2^{n-1} - 1$ // agrupa 2^n-1
T(n) = $2 * 2^{n-1} - 1$ //retira a multiplicação 2 por -1
T(n) = $2^n - 1$

8) Logo a complexidade da fórmula,
T(n)= $2T(n - 1) + O(1)$ ϵ $\Theta(2^n)$

9) Prova por indução,
Passo base: para n = 1, o resultado esperado é 0
T(n) = $2^n - 1$
T(1) = $2^1 - 1$
T(1) = $2 - 1$
T(1) = 1 (correto)

Passo indutivo: por hipótese de indução, assumimos que a fórmula está correta para n-1, isto é, T(n-1) = $2^{(n-1)} - 1$. Então, temos que verificar se T(n) = $2^n - 1$, sabendo-se que T(n) = $2T(n-1) + 1$ e partindo da H.I. que T(n-1) = $2^{(n-1)} - 1$
T(n) = $2T(n-1) + 1$
T(n) = 2$(2^{(n-1)} - 1) + 1$ //Multiplica por 2
T(n) = $(2 * 2^{(n-1)} - 2) + 1$ // Simplifique retirando 2 e subtraindo 1 do expoente
T(n) = $2^n - 1$ (passo indutivo provado)

Demonstrado que 2T(n - 1) + 1 = $2^n - 1$ para n > 1

$(c)\ \mathbf{T(n) = 9T\left(\frac{n}{3}\right) + \boldsymbol{n^2}}$
R.:
a=9
b=3
k=2 //Expoente de n
f(n) = n^2
$\Theta(n^{log_b a})$= $\Theta(n^{log_3 9})$=$\Theta(n^2)$ //$log_3 9 = 2$

Comparação #1:
f(n) ∈ O(($n^{log_b a - \varepsilon}$))
n^2= $n^{log_b a - \varepsilon}$)
2 = $log_3 9 - \varepsilon$)
2=$(2 - \varepsilon)$
$\varepsilon = 2 - 2$
$\varepsilon = 0$ (Atende o caso 2)

Comparação #2:
f(n) = $\Theta(n^{log_b a})$ ou seja,
n^2= $n^{log_3 9}$
n^2= n^2 // É igual portanto é o Caso 2 do Teorema Mestre
$log_b a < k$ //Caso 3
$log_3 9 < 2$
2< 2 //Portanto não é o caso 3
$log_b a > k$ //Caso 1
$log_3 9 > 2$
2> 2 //Portanto não é o caso 1

Temos o Caso 2 do Teorema Mestre, portanto
$\mathrm{T(n)} = 9\mathrm{T}\left(\frac{\mathrm{n}}{3}\right) + n^2$
$\mathrm{T(n)} = \Theta(n^{log_3 9}\ log_2 n)$
$\mathrm{T(n)} = \Theta(n^2\ log_2 n)$
Então T(n)∈ $\Theta(n^2\ log_2 n\ )$

Teorema 3.1 do livro do Cormen, tradução da 3ª edição americana:

"Para quaisquer duas funções f(n) e g(n), temos f(n)=(g(n)) se e somente se f(n)=O(g(n)) e f(n)=(g(n))".

Pode-se, portanto, concluir que, se f(n)=(g(n)), logo f(n)=O(g(n)) e f(n)=(g(n)).

Letra	**Problema**	**Consumo de Tempo (Complexidade)**		
		Θ	**Ω(Melhor Caso)**	**O(Pior Caso)**
a	$T(n) = 5T\left(\frac{n}{2}\right) + \mathrm{n}$	T(n)∈ $\Theta\left(n^{log_2 5}\right)$	$\Omega(n^{log_2 5})$	$O(n^{log_2 5})$
b	$T(n) = 2T(n-1) + 1$	T(n)∈ $\Theta(2^n)$	$\Omega(2^n)$	$O(2^n)$
c	$\mathrm{T(n)} = 9\mathrm{T}\left(\frac{\mathrm{n}}{3}\right) + n^2$	T(n)∈ $\Theta(n^2\ log_2\ n\ )$	$\Omega(n^2\ log_2\ n)$	$O(n^2\ log_2\ n)$

Qual o consumo de tempo de cada um desses algoritmos?
R: O consumo de tempo está diretamente ligado à complexidade de cada algoritmo, dos recursos computacionais onde o mesmo será executado e o tamanho da entrada a ser processada. Nesta questão especificamente, apenas podemos nos basear nas suas respectivas complexidades, apresentadas na tabela acima, como indicadores de consumo de tempo.

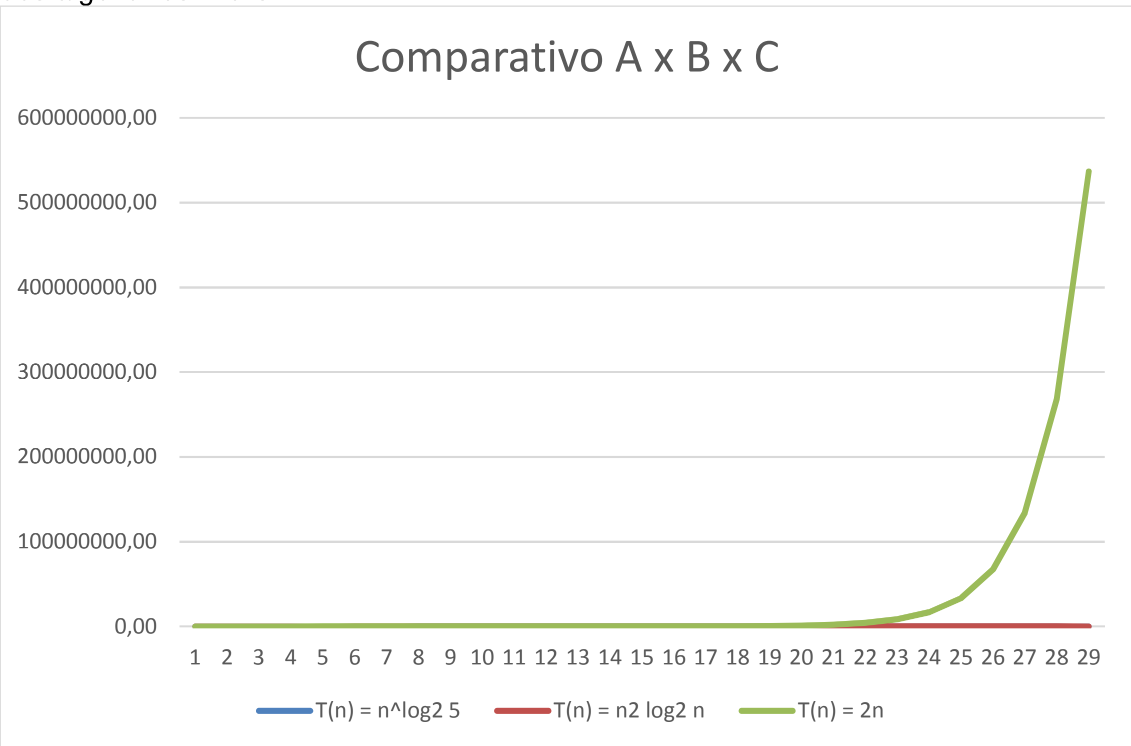
Qual algoritmo é assintoticamente mais eficiente no pior caso?
R. O algoritmo A é mais eficiente assintoticamente pois sua complexidade, tanto para o melhor(Ω) quando para o pior caso (O), é $n^{log_2 5}$. Na comparação com os algoritmos B e C, o algoritmo A possui um melhor desempenho computacional.

Abaixo a tabela com a simulação dos valores de 1 a 30 para n

n	**T(n) = $\mathbf{n^{log\ 5}}$**	**T(n) = $\mathbf{2^n}$**	**T(n) = $\mathbf{n^2\ log_2\ n}$**
1	1,00	2,00E+00	0,00
2	5,00	4,00E+00	4,00
3	12,82	8,00E+00	14,26
4	25,00	1,60E+01	32,00
5	41,97	3,20E+01	58,05
6	64,09	6,40E+01	93,06
7	91,68	1,28E+02	137,56
8	125,00	2,56E+02	192,00
9	164,32	5,12E+02	256,76
10	209,86	1,02E+03	332,19
11	261,84	2,05E+03	418,59
12	320,47	4,10E+03	516,23
13	385,92	8,19E+03	625,37
14	458,38	1,64E+04	746,24
15	538,02	3,28E+04	879,05
16	625,00	6,55E+04	1024,00
17	719,47	1,31E+05	1181,28
18	821,58	2,62E+05	1351,06
19	931,48	5,24E+05	1533,50
20	1049,30	1,05E+06	1728,77
21	1175,16	2,10E+06	1937,01

22	1309,21	4,19E+06	2158,36
23	1451,56	8,39E+06	2392,96
24	1602,33	1,68E+07	2640,94
25	1761,64	3,36E+07	2902,41
26	1929,60	6,71E+07	3177,50
27	2106,32	1,34E+08	3466,31
28	2291,91	2,68E+08	3768,97
29	2486,47	5,37E+08	4085,56
30	2690,11	1,07E+09	4416,20

Construindo um gráfico, percebe que o algoritmo B tem u desempenho muito ruim em comparado aos algoritmos A e C. O próximo gráfico contém a comparação dos dados dos algoritmos A e C.



Comparando separadamente os algoritmos A e C percebe que o algoritmo A é mais eficiente em relação a C.