Ordenação externa

ALGORITMOS E ESTRUTURAS DE DADOS III

Prof. Marcos André S. Kutova

Ordenação externa

 Processo de ordenação com auxílio de algum dispositivo de memória externa (HD, fitas, ...), adotado quando os dados a serem ordenados são maiores que a capacidade de ordenação em memória principal.

Objetivo:

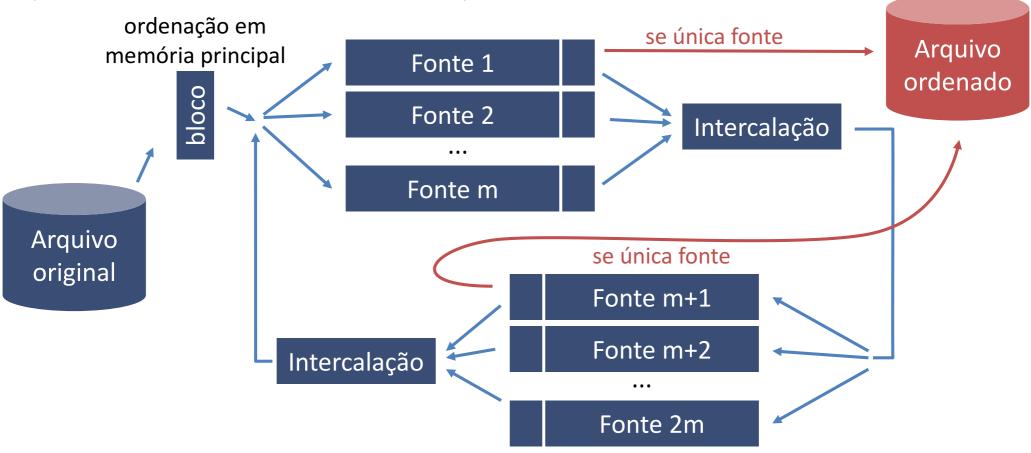
 Ordenar os registros com o menor número de leituras e escritas em disco

 Algoritmo que ordena os registros por meio da intercalação de N caminhos balanceados (arquivos com tamanho aproximado)

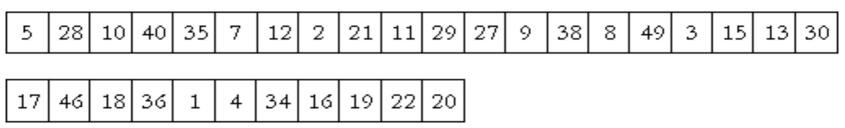
• Etapas:

- Distribuição de blocos de registros ordenados por N caminhos balanceados
- 2. Intercalações sucessivas dos N caminhos

Intercalação balanceada (de m caminhos)



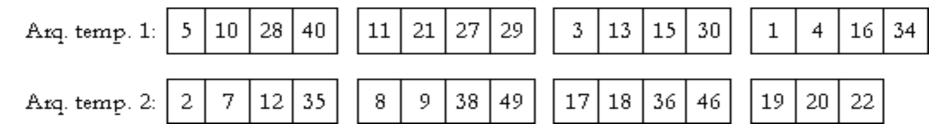
Arquivo original



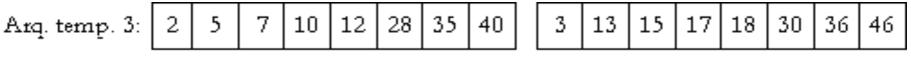
• Distribuição de blocos ordenados de tamanho 4 por 2 fontes

Arq. temp. 1: Arq. temp. 2:

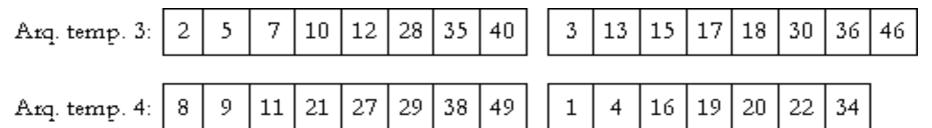
Primeira intercalação



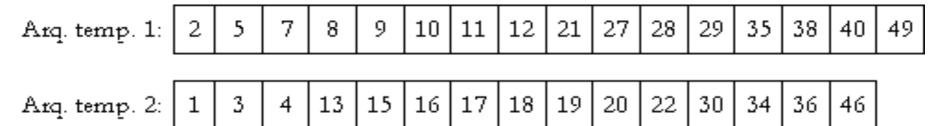
A intercalação de 2 blocos ordenados de tamanho 4 gera um bloco ordenado de tamanho 8



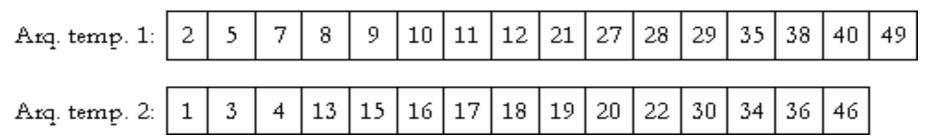
Segunda intercalação



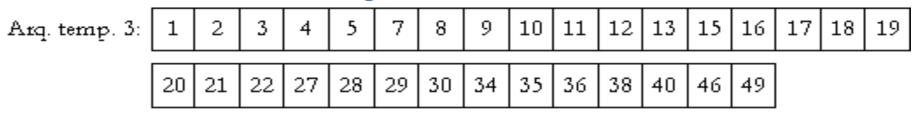
A intercalação de 2 blocos ordenados de tamanho 8 gera um bloco ordenado de tamanho 16



Terceira intercalação



A intercalação de 2 blocos ordenados de tamanho 16 gera um bloco ordenado de tamanho 32



Arq. temp. 4:

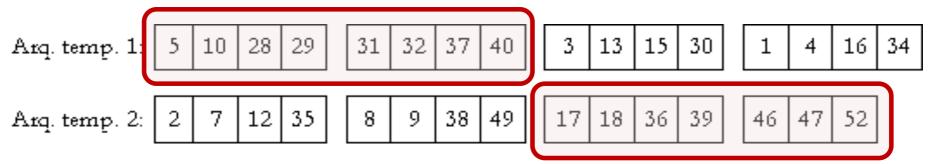
Quando todos os registros ficarem em uma única fonte, o arquivo estará ordenado.

Análise de complexidade

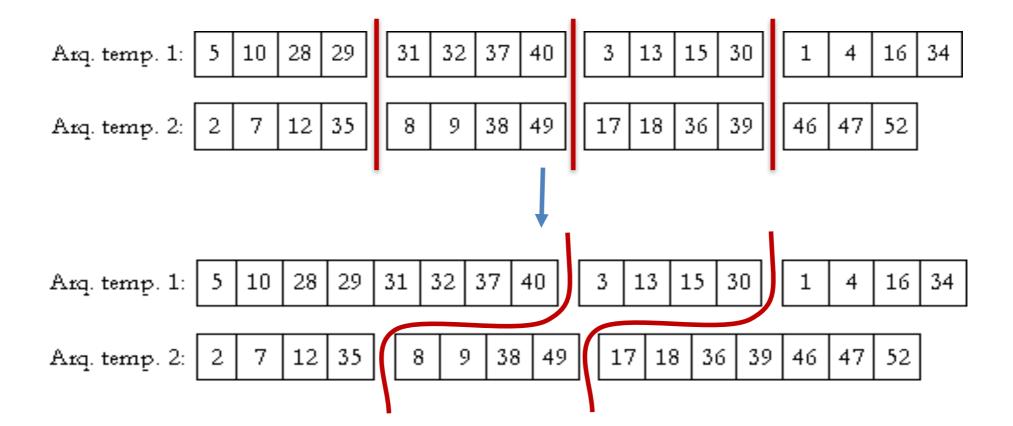
- A base da análise é o número de "passadas" (leituras/escritas) pelos registros
- O tempo da ordenação em memória principal é pouco significativo
- Estimativa:
 - $passadas = log_m \left(\frac{N}{h}\right)$
 - Em que:
 - N = número total de registros
 - b = tamanho do bloco ordenado em memória principal
 - m = quantidade de fontes usadas na intercalação

Segmentos de tamanho variável

 Uma primeira estratégia de otimização é tentar aproveitar a eventual ordenação entre os blocos:



Segmentos de tamanho variável

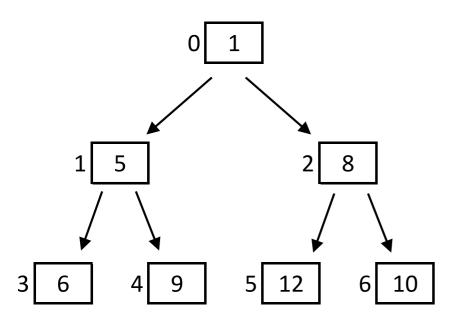


Intercalação com seleção por substituição

- A segunda estratégia de otimização é de gerar segmentos ordenados maiores na fase de distribuição
- Requer o uso de uma fila de prioridades, como um heap.

Heap de mínimo

()	1	2	3	4	5	6
	1	5	8	6	9	12	10



filhoesquerdo(i) = i * 2 + 1

filhodireito(i) = i * 2 + 2

$$pai(i) = int(\frac{i-1}{2})$$

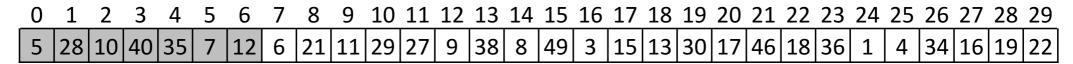
Operações:

- Inserir novo elemento (na raiz e afundá-lo)
- Remover a raiz (colocar o último elemento lá e afundá-lo)

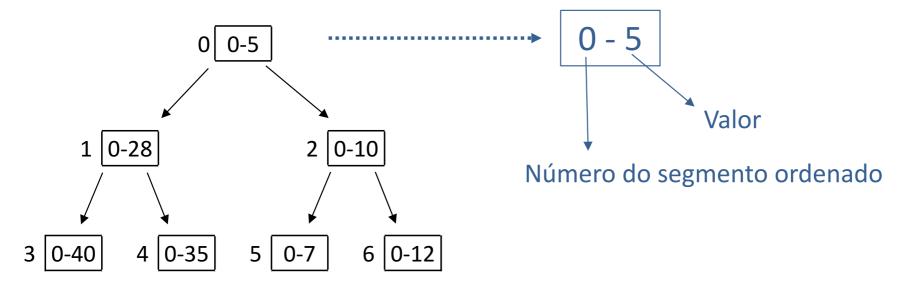
Uso de heap na intercalação

Arquivo a ser ordenado:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
5	28	10	40	35	7	12	6	21	11	29	27	9	38	8	49	3	15	13	30	17	46	18	36	1	4	34	16	19	22

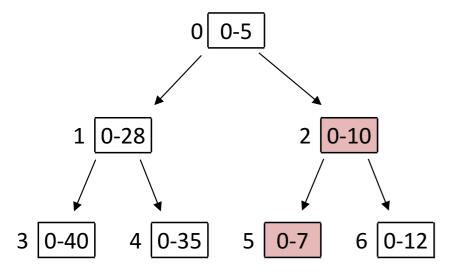


Passo 1: Preenchimento do heap



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
5	28	10	40	35	7	12	6	21	11	29	27	9	38	8	49	3	15	13	30	17	46	18	36	1	4	34	16	19	22

Passo 2: Organização do *heap*



Da posição (n-1)/2, que representa o primeiro pai, até a posição 0, afundar cada elemento.

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Retirada do menor elemento

 0
 1
 2
 3
 4
 5
 6

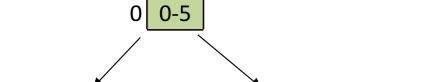
 0-5
 0-28
 0-7
 0-40
 0-35
 0-10
 0-12

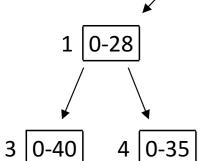
Saídas:

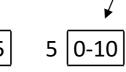
Arq.1:

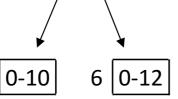
5

Arq.2:









0-7

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Inserção do próximo registro

 0
 1
 2
 3
 4
 5
 6

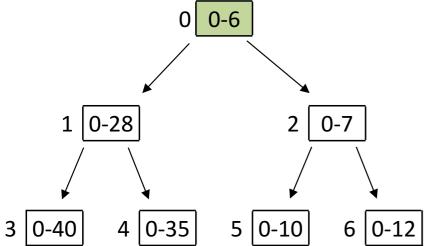
 0-6
 0-28
 0-7
 0-40
 0-35
 0-10
 0-12

Saídas:

Arq.1:

5





 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Inserção do próximo registro

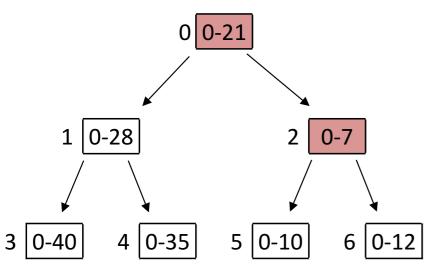
 0
 1
 2
 3
 4
 5
 6

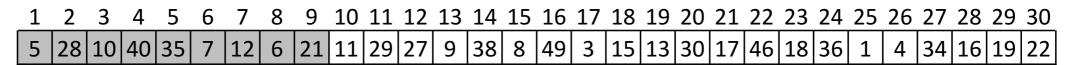
 0-21
 0-28
 0-7
 0-40
 0-35
 0-10
 0-12

Saídas:

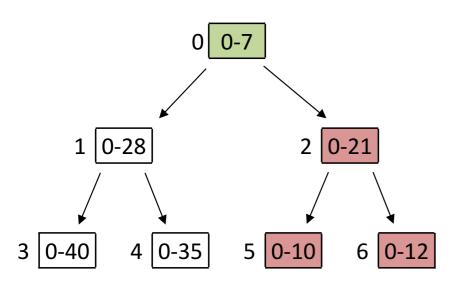
Arq.1:

5 6





Inserção do próximo registro

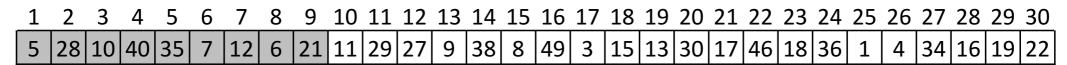


Saídas:

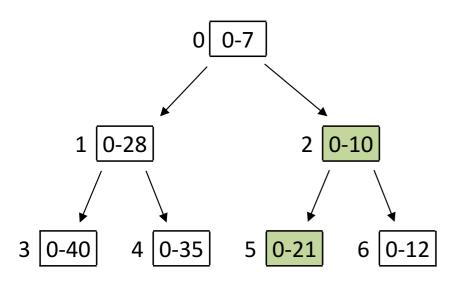
Arq.1:

5 6





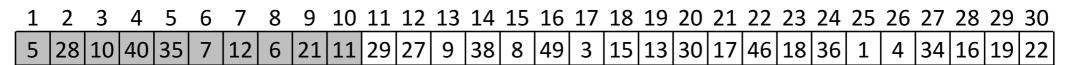
Inserção do próximo registro



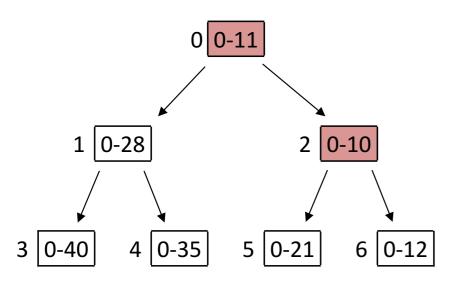
Saídas:

Arq.1:

5 6



Inserção do próximo registro

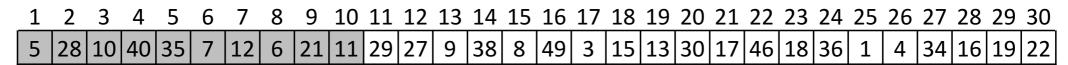


Saídas:

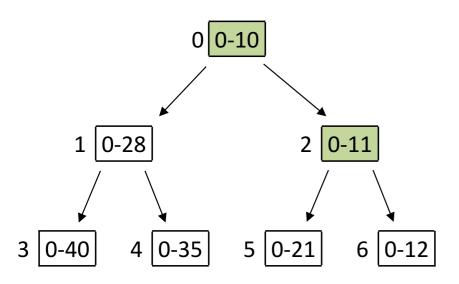
Arq.1:

5 6 7





Inserção do próximo registro

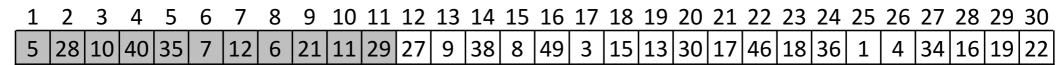


Saídas:

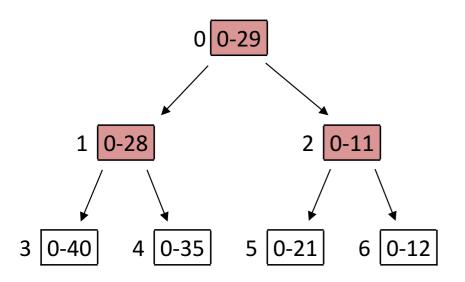
Arq.1:

5 6 7





Inserção do próximo registro

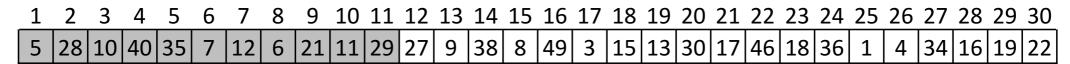


Saídas:

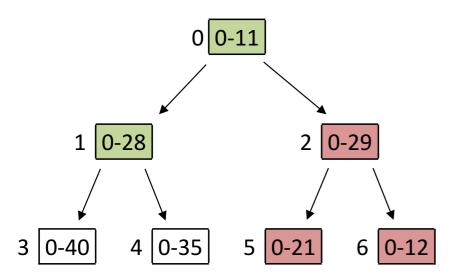
Arq.1:

5 6 7 10





Inserção do próximo registro

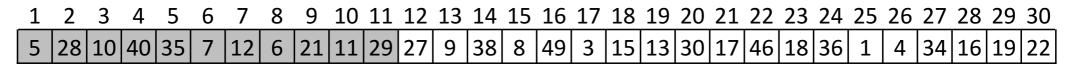


Saídas:

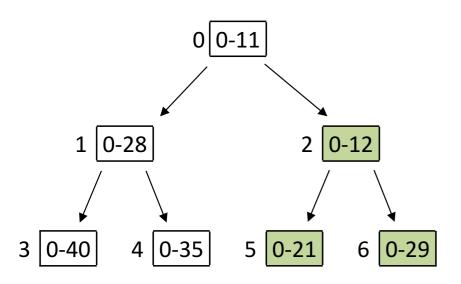
Arq.1:

5 6 7 10





Inserção do próximo registro

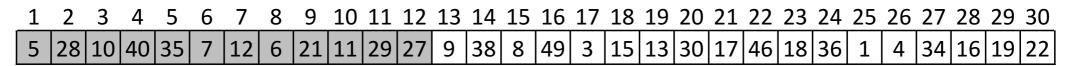


Saídas:

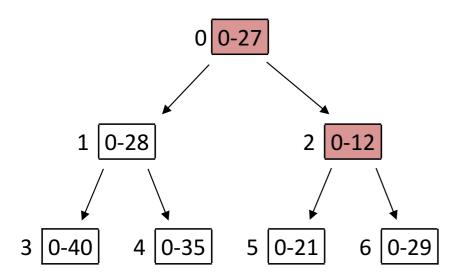
Arq.1:

5 6 7 10





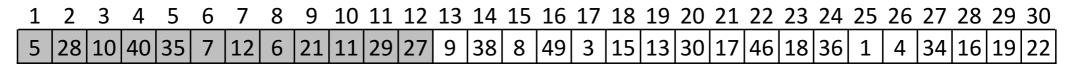
Inserção do próximo registro



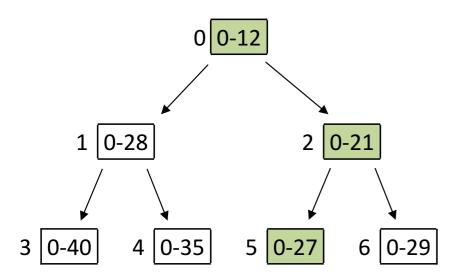
Saídas:

Arq.1:





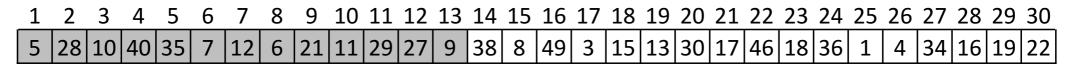
Inserção do próximo registro



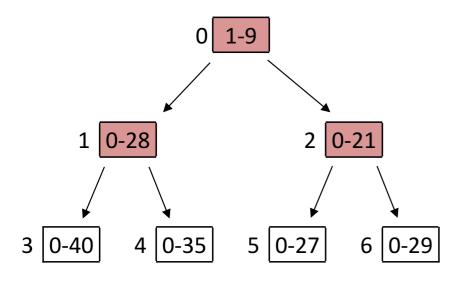
Saídas:

Arq.1:





Inserção do próximo registro

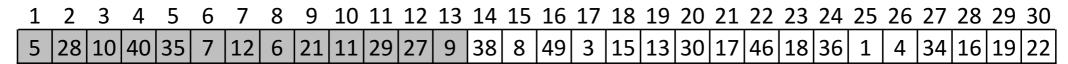


Saídas:

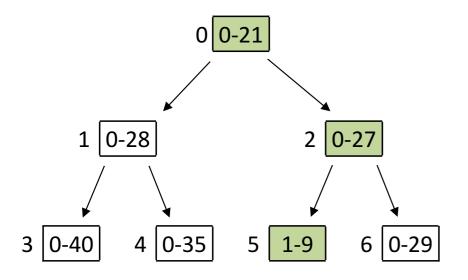
Arq.1: 5 6 7 10 11 12

Arq.2:

9 < 12 Se ele entrar aqui, o segmento ordenado é finalizado



Inserção do próximo registro

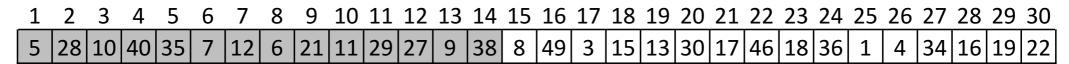


Saídas:

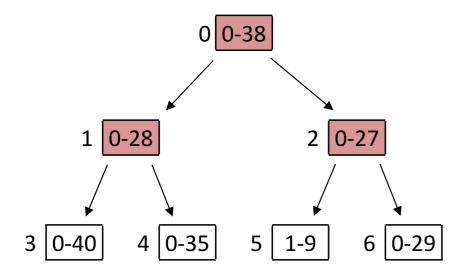
Arq.1:

5 6 7 10 11 12





Inserção do próximo registro



Saídas:

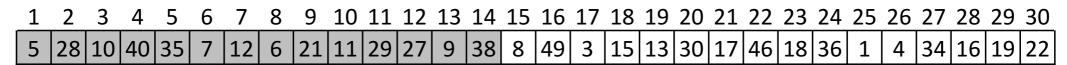
Arq.1:

5 | 6 | 7 | 10 | 11 | 12 | 21

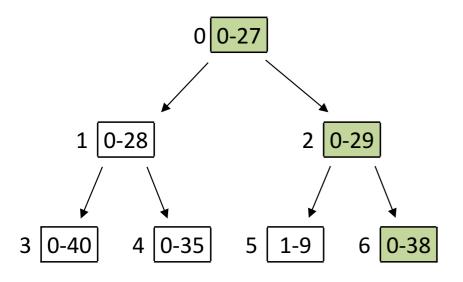
Arq.2:

38 > 21

Ele pode entrar aqui sem quebrar a ordenação do segmento



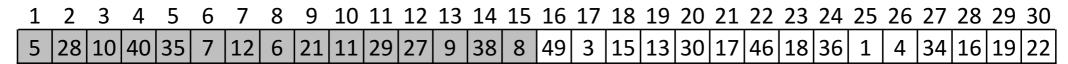
Inserção do próximo registro



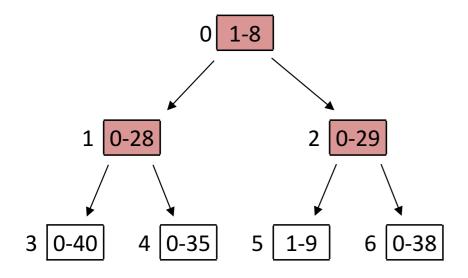
Saídas:

Arq.1:



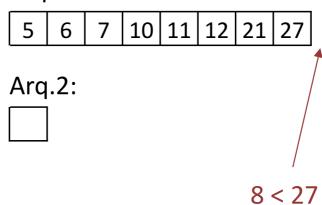


Inserção do próximo registro

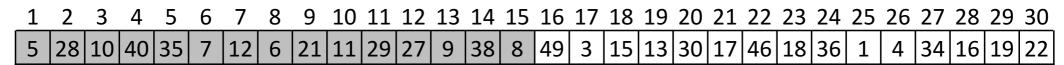


Saídas:

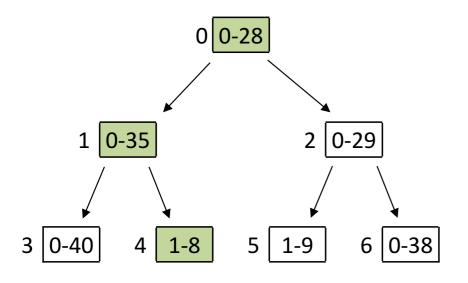
Arq.1:



Se ele entrar aqui, o segmento ordenado é finalizado



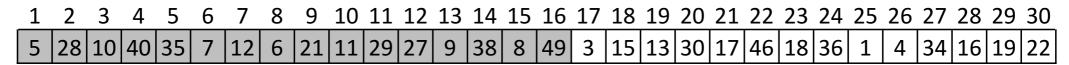
Inserção do próximo registro



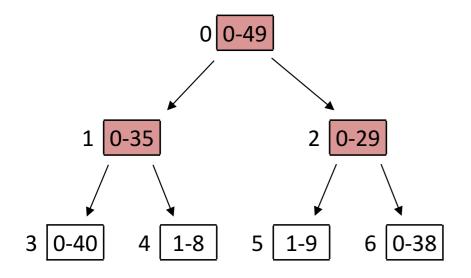
Saídas:

Arq.1:



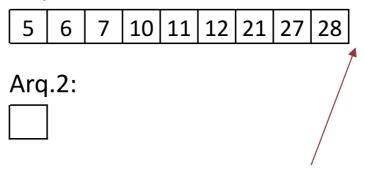


Inserção do próximo registro

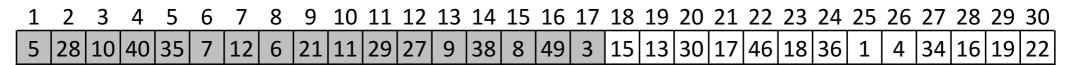


Saídas:

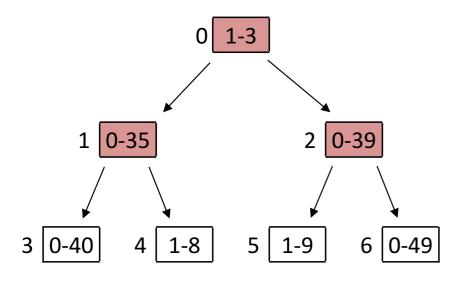
Arq.1:



49 > 28
Ele pode entrar aqui
sem quebrar a
ordenação do segmento



Inserção do próximo registro

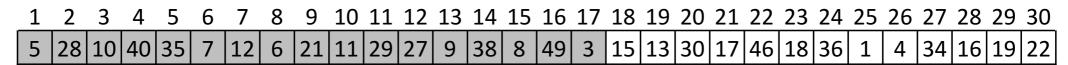


Saídas:

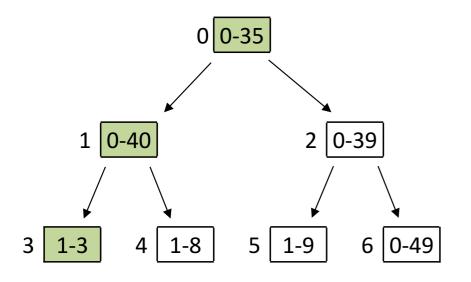
Arq.1:

5	6	7	10	11	12	21	27	28	29
---	---	---	----	----	----	----	----	----	----





Inserção do próximo registro

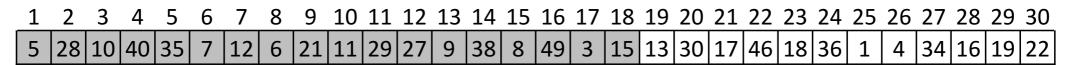


Saídas:

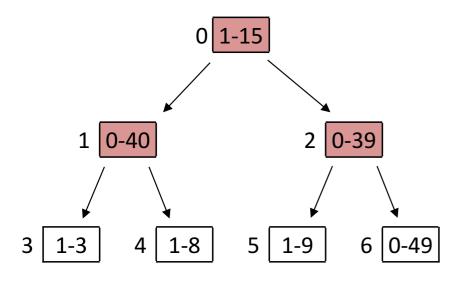
Arq.1:

5	6	7	10	11	12	21	27	28	29
---	---	---	----	----	----	----	----	----	----





Inserção do próximo registro

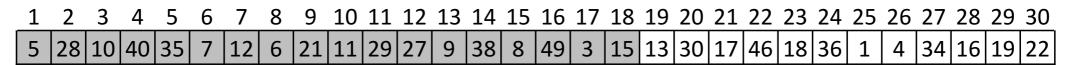


Saídas:

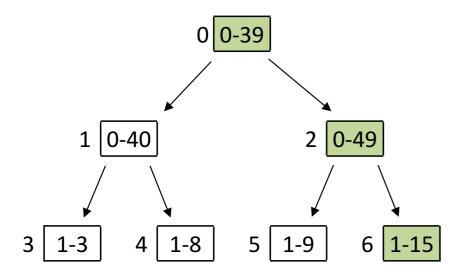
Arq.1:

5	6	7	10	11	12	21	27	28	29	35
---	---	---	----	----	----	----	----	----	----	----





Inserção do próximo registro

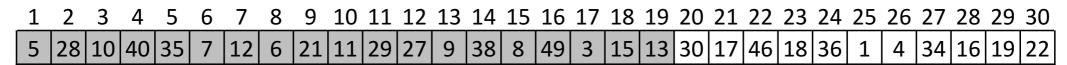


Saídas:

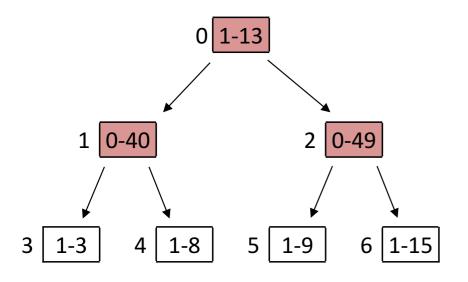
Arq.1:

5	6	7	10	11	12	21	27	28	29	35
---	---	---	----	----	----	----	----	----	----	----





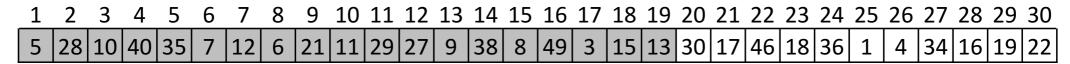
Inserção do próximo registro



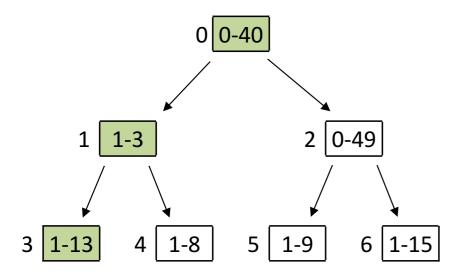
Saídas:

Arq.1:





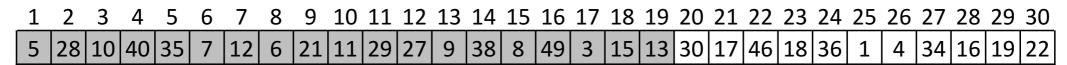
Inserção do próximo registro



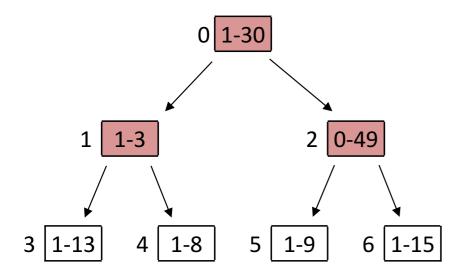
Saídas:

Arq.1:





Inserção do próximo registro

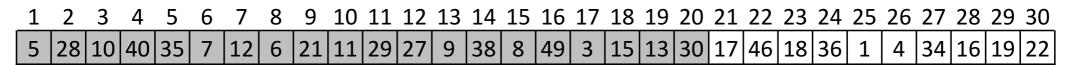


Saídas:

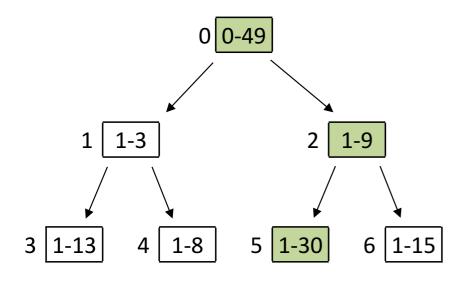
Arq.1:

5	(5	7	10	11	12	21	27	28	29	35	39	40
---	---	---	---	----	----	----	----	----	----	----	----	----	----





Inserção do próximo registro

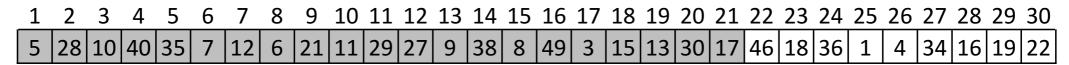


Saídas:

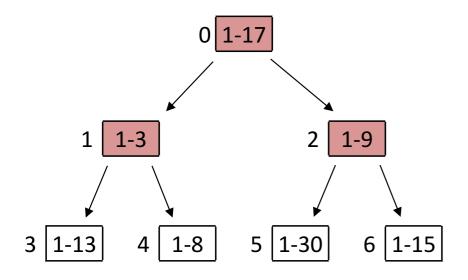
Arq.1:

	5	6	7	10	11	12	21	27	28	29	35	39	40
--	---	---	---	----	----	----	----	----	----	----	----	----	----





Inserção do próximo registro

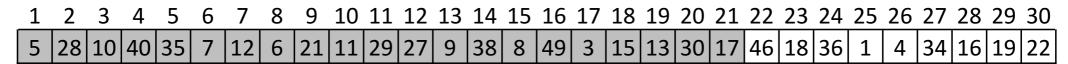


Saídas:

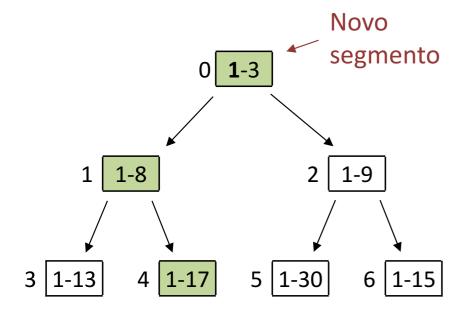
Arq.1:

5	6	7	10	11	12	21	27	28	29	35	39	40	49
---	---	---	----	----	----	----	----	----	----	----	----	----	----





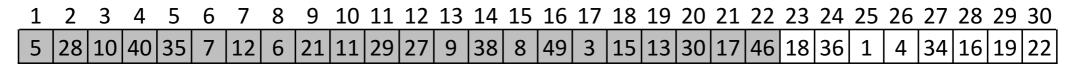
Inserção do próximo registro



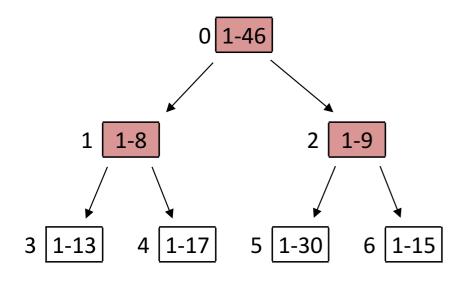
Saídas:

Arq.1:





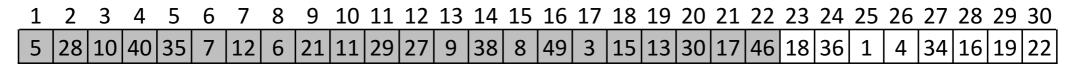
Inserção do próximo registro



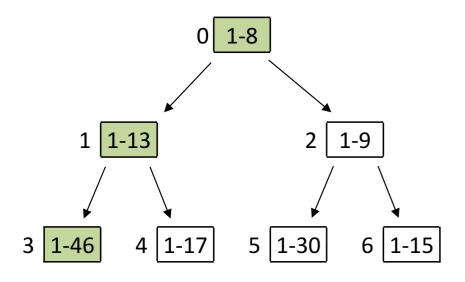
Saídas:

Arq.1:

Arq.2:



Inserção do próximo registro

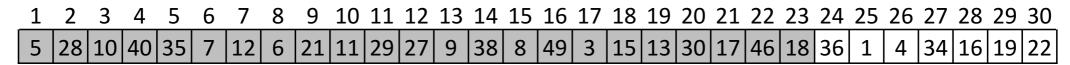


Saídas:

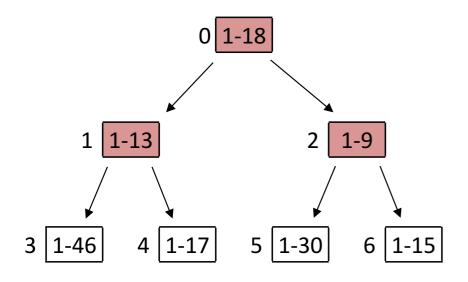
Arq.1:

5	6	7	10	11	12	21	27	28	29	35	39	40	49
---	---	---	----	----	----	----	----	----	----	----	----	----	----

Arq.2:



Inserção do próximo registro

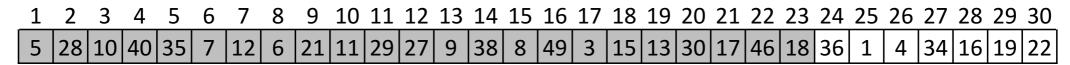


Saídas:

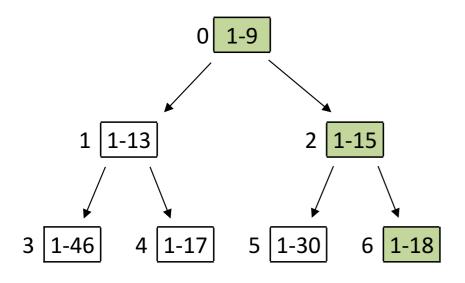
Arq.1:

5	6	7	10	11	12	21	27	28	29	35	39	40	49
---	---	---	----	----	----	----	----	----	----	----	----	----	----

Arq.2:



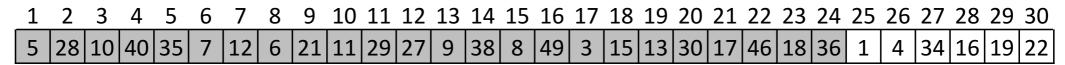
Inserção do próximo registro



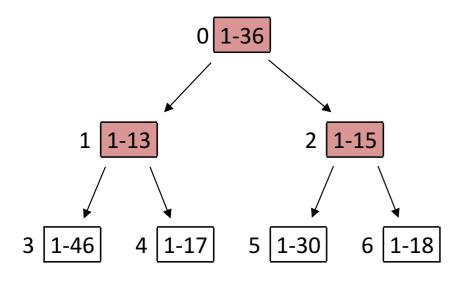
Saídas:

Arq.1:

Arq.2:



Inserção do próximo registro



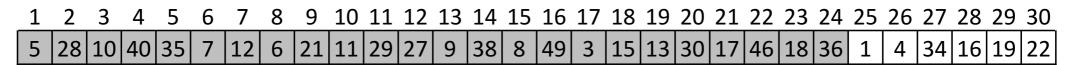
Saídas:

Arq.1:

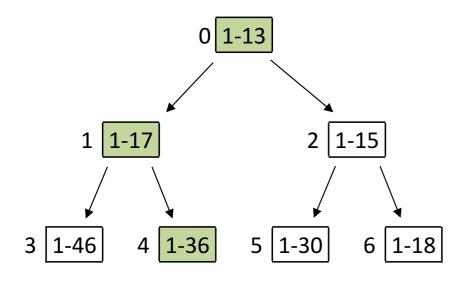
5	6	7	10	11	12	21	27	28	29	35	39	40	49
---	---	---	----	----	----	----	----	----	----	----	----	----	----

Arq.2:

3 | 8 | 9



Inserção do próximo registro



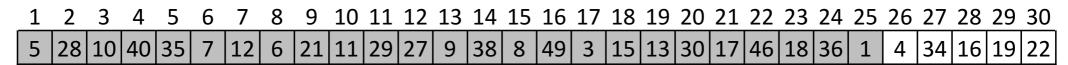
Saídas:

Arq.1:

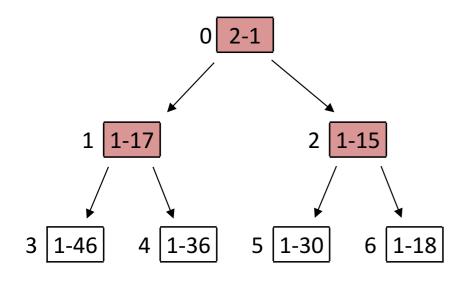
	5	6	7	10	11	12	21	27	28	29	35	39	40	49
--	---	---	---	----	----	----	----	----	----	----	----	----	----	----

Arq.2:

3 8 9



Inserção do próximo registro



Saídas:

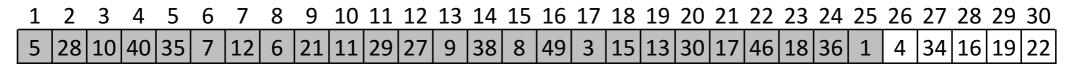
Arq.1:

5 6 7 10 11 12 21 27 28 29 35 39 40 49

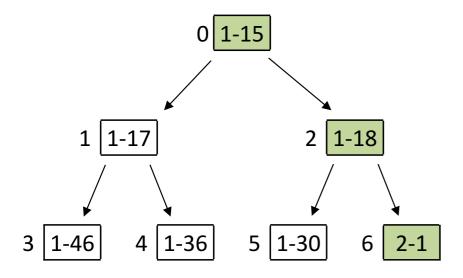
Arq.2:

3 8 9 13

1 < 13 Se ele entrar aqui, o segmento ordenado é finalizado



Inserção do próximo registro



Saídas:

Arq.1:

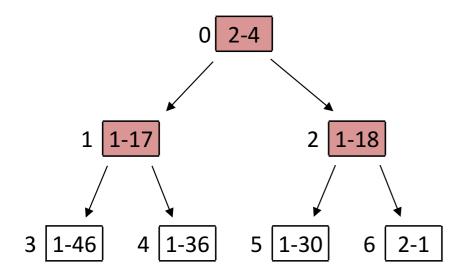
Arq.2:

3 8 9 13

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Inserção do próximo registro



Saídas:

Arq.1:

5 6 7 10 11 12 21 27 28 29 35 39 40 49

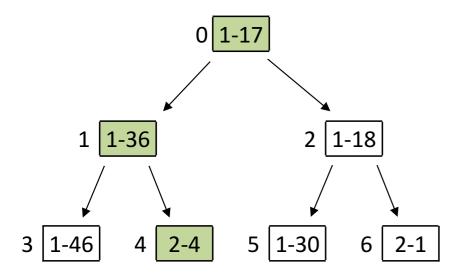
Arq.2:

3 8 9 13 15

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Inserção do próximo registro



Saídas:

Arq.1:

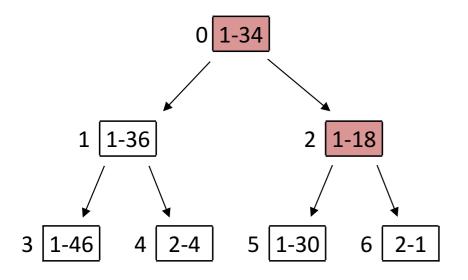
Arq.2:

3 8 9 13 15

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Inserção do próximo registro



Saídas:

Arq.1:

5 6 7 10 11 12 21 27 28 29 35 39 40 49

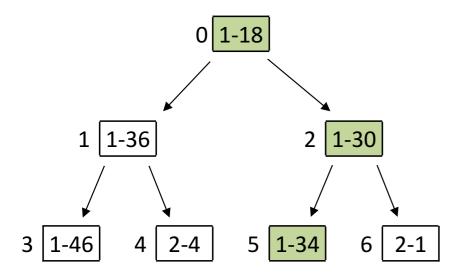
Arq.2:

3 8 9 13 15 17

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Inserção do próximo registro



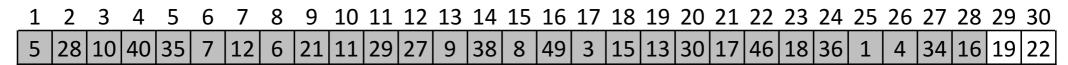
Saídas:

Arq.1:

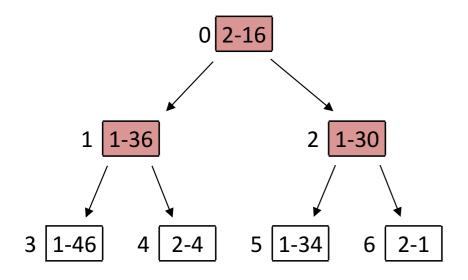
5 6 7 10 11 12 21 27 28 29 35 39 40 49

Arq.2:

3 8 9 13 15 17



Inserção do próximo registro



Saídas:

Arq.1:

5 6 7 10 11 12 21 27 28 29 35 39 40 49

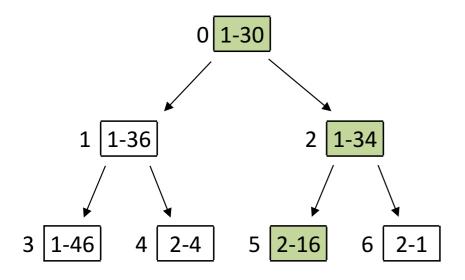
Arq.2:

3 8 9 13 15 17 18

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

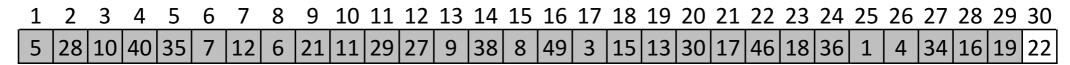
Inserção do próximo registro



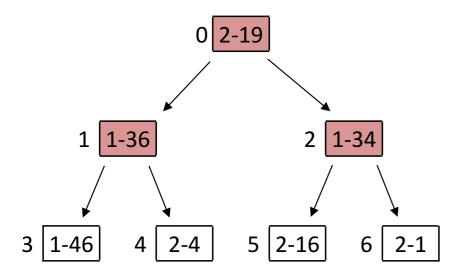
Saídas:

Arq.1:

3 8 9 13 15 17	18
----------------	----



Inserção do próximo registro



Saídas:

Arq.1:

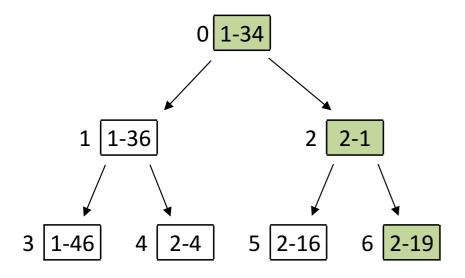
Arq.2:

3 | 8 | 9 | 13 | 15 | 17 | 18 | 30

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Inserção do próximo registro



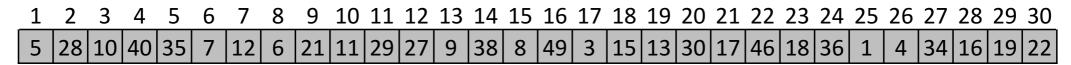
Saídas:

Arq.1:

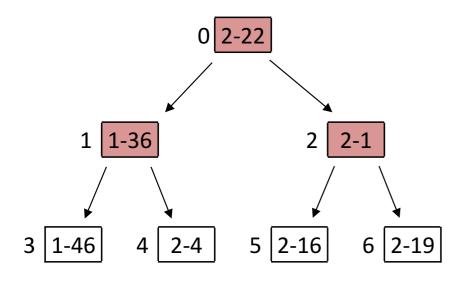
5 6 7 10 11 12 21 27 28 29 35 39 40 49

Arq.2:

3 | 8 | 9 | 13 | 15 | 17 | 18 | 30



Inserção do próximo registro



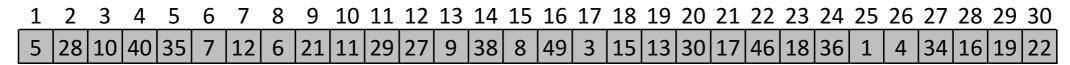
Saídas:

Arq.1:

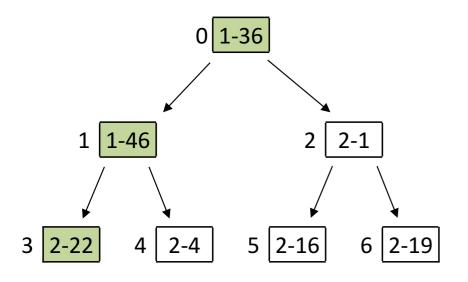
5	6	7	10	11	12	21	27	28	29	35	39	40	49
---	---	---	----	----	----	----	----	----	----	----	----	----	----

Arq.2:

3 8 9 13 15 17 18 30 34



Inserção do próximo registro



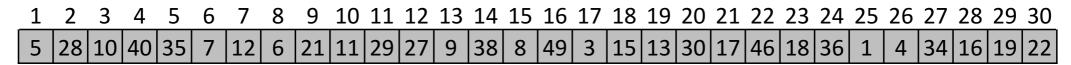
Saídas:

Arq.1:

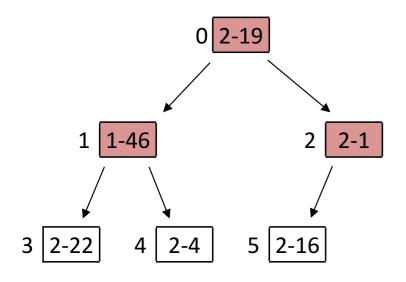
5 6 7 10 11 12 21 27 28 29 35 39 40 49

Arq.2:

3 | 8 | 9 | 13 | 15 | 17 | 18 | 30 | 34



Esvaziamento do heap

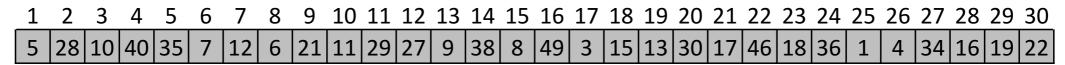


Saídas:

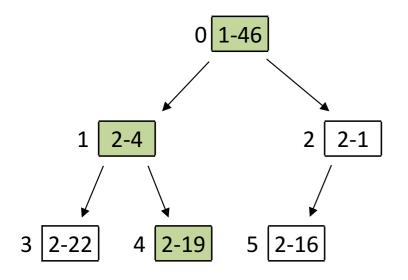
Arq.1:

5 6 7 10 11 12 21 27 28 29 35 39 40 49

Arq.2:



Esvaziamento do heap

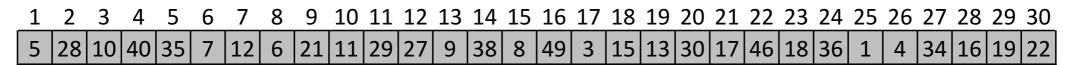


Saídas:

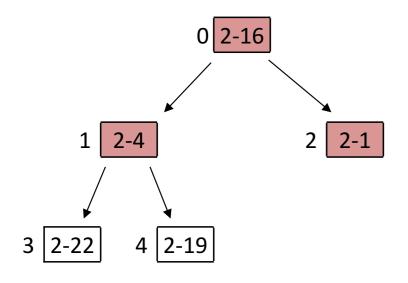
Arq.1:

5 | 6 | 7 | 10 | 11 | 12 | 21 | 27 | 28 | 29 | 35 | 39 | 40 | 49

Arq.2:



Esvaziamento do heap

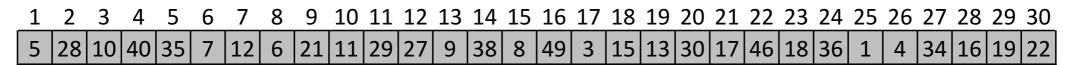


Saídas:

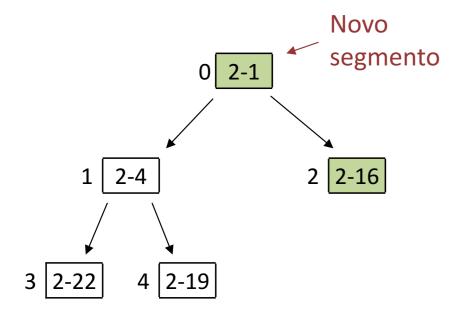
Arq.1:

5 6 7 10 11 12 21 27 28 29 35 39 40 49

Arq.2:



Esvaziamento do heap

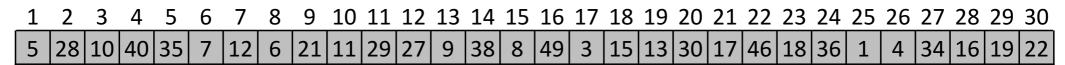


Saídas:

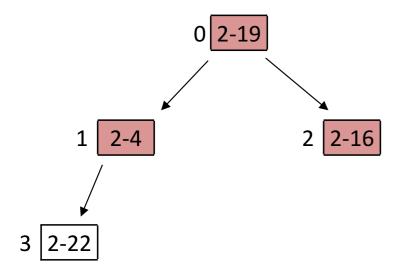
Arq.1:

5	6	7	10	11	12	21	27	28	29	35	39	40	49
---	---	---	----	----	----	----	----	----	----	----	----	----	----

3	8	9	13	15	17	18	30	34	36	46
---	---	---	----	----	----	----	----	----	----	----



Esvaziamento do heap



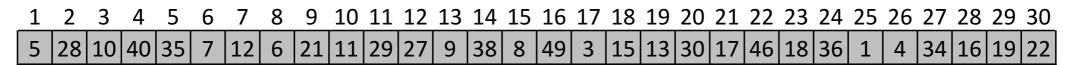
Saídas:

Arq.1:

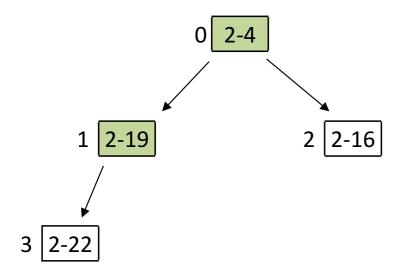
5 6 7 10 11 12 21 27 28 29 35 39 40 49 1

Arq.2:

3 | 8 | 9 | 13 | 15 | 17 | 18 | 30 | 34 | 36 | 46



Esvaziamento do heap



Saídas:

Arq.1:

5 6 7 10 11 12 21 27 28 29 35 39 40 49 1

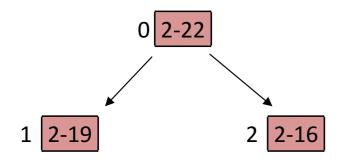
Arq.2:

3 | 8 | 9 | 13 | 15 | 17 | 18 | 30 | 34 | 36 | 46

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

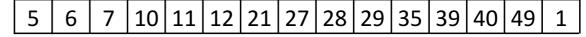
 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Esvaziamento do heap



Saídas:

Arq.1:



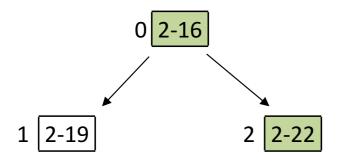
4

Arq.2:

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

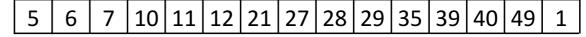
 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Esvaziamento do heap



Saídas:

Arq.1:



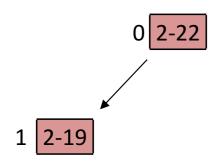
4

Arq.2:

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

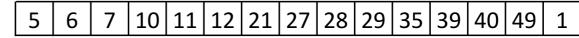
 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Esvaziamento do heap



Saídas:

Arq.1:



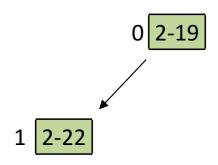
4 16

Arq.2:

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

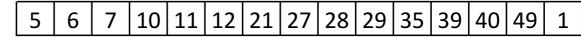
 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Esvaziamento do heap



Saídas:

Arq.1:



4 16

Arq.2:

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Esvaziamento do heap

0 2-22

Saídas:

Arq.1:

4 | 16 | 19 |

Arq.2:

 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30

 5
 28
 10
 40
 35
 7
 12
 6
 21
 11
 29
 27
 9
 38
 8
 49
 3
 15
 13
 30
 17
 46
 18
 36
 1
 4
 34
 16
 19
 22

Saídas:

Arq.1:

Arq.2:

Agora é só fazer as intercalações...