

Estrutura de Dados II

Prof. Me. Pietro M. de Oliveira



Bubblesort

Ordenação por flutuação ou Método da bolha





Também conhecido como técnica de **ordenação por flutuação**

Simples implementação

Alto custo computacional

Estável

Compara todos os elementos entre si

Ineficiente, independentemente da distribuição:

- Ordeando
- Parcialmente ordenado
- Não ordenado





Algoritmo

- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca *A[i] ↔ A[j]*



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

<u> </u>	j								
0	1	2	3	4	5	6	7	8	9
26	32	46	19	15	67	81	22	55	01



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca *A[i]* ↔ *A[j]*

	j								
0	1	2	3	4	5	6	7	8	9
26	32	46	19	15	67	81	22	55	01



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

i		j								
0	1	2	3	4	5	6	7	8	9	
26	32	46	19	15	67	81	22	55	01	



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca *A[i]* ↔ *A[j]*

i		j								
0	1	2	3	4	5	6	7	8	9	
26	32	46	19	15	67	81	22	55	01	



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

i			j							
0	1	2	3	4	5	6	7	8	9	
26	32	46	19	15	67	81	22	55	01	1



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

j			j						
0	1	2	3	4	5	6	7	8	9
19	32	46	26	15	67	81	22	55	01



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

				j						
0	1	2	3	4	5	6	7	8	9	
19	32	46	26	15	67	81	22	55	01	1



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

				j					
0	1	2	3	4	5	6	7	8	9
15	32	46	26	19	67	81	22	55	01



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

					<u>j</u>					
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

j					j					
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	2



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

i						j			
0	1	2	3	4	5	6	7	8	9
15	32	46	26	19	67	81	22	55	01



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i] ↔ A[j]*

j						j				
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	2



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

i							j			
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	2



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca *A[i]* ↔ *A[j]*

İ							j			
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	24



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

i								j		
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	2



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca *A[i]* ↔ *A[j]*

İ								j		
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	2



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

j									j	
0	1	2	3	4	5	6	7	8	9	
15	32	46	26	19	67	81	22	55	01	2



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

j									j	
0	1	2	3	4	5	6	7	8	9	
01	32	46	26	19	67	81	22	55	15	28



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

_	

0	1	2	3	4	5	6	7	8	9
01	32	46	26	19	67	81	22	55	15



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca *A[i]* ↔ *A[j]*

	i	j							
0	1	2	3	4	5	6	7	8	9
01	32	46	26	19	67	81	22	55	15



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i		j							
0	1	2	3	4	5	6	7	8	9	
01	32	46	26	19	67	81	22	55	15	



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i		j						
0	1	2	3	4	5	6	7	8	9
01	26	46	32	19	67	81	22	55	15



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i			j						
0	1	2	3	4	5	6	7	8	9	
01	26	46	32	19	67	81	22	55	15	3



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i			j					
0	1	2	3	4	5	6	7	8	9
01	19	46	32	26	67	81	22	55	15



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

	i				j					
0	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	3



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

					j					
O	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i					j				
0	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	3



38

- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i					j				
0	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	,



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

	i						j			
0	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	3



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca $A[i] \leftrightarrow A[j]$

	i						j			
0	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	4

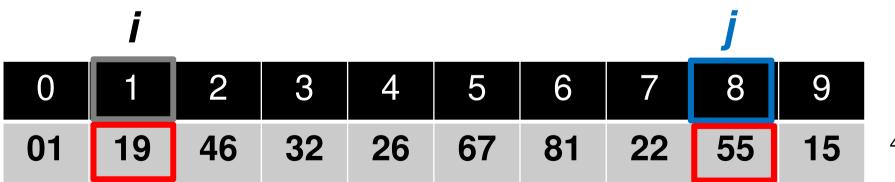


- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i							j		
0	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	4



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se **A[i] > A[j]** então
- 4. troca $A[i] \leftrightarrow A[j]$





- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

	i								j	
0	1	2	3	4	5	6	7	8	9	
01	19	46	32	26	67	81	22	55	15	43



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

	i								j	
0	1	2	3	4	5	6	7	8	9	
01	15	46	32	26	67	81	22	55	19	44



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

-		_

0	1	2	3	4	5	6	7	8	9
01	15	46	32	26	67	81	22	55	19



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

		i	j							
0	1	2	3	4	5	6	7	8	9	
01	15	32	56	26	67	81	22	55	19	4



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

		i		<u>j</u>						
0	1	2	3	4	5	6	7	8	9	
01	15	32	56	26	67	81	22	55	19	47



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

		i		j						
0	1	2	3	4	5	6	7	8	9	
01	15	26	56	32	67	81	22	55	19	•



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

		i			j					
0	1	2	3	4	5	6	7	8	9	
01	15	26	56	32	67	81	22	55	19	4



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

		i			j					
0	1	2	3	4	5	6	7	8	9	
01	15	26	56	32	67	81	22	55	19	į



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

		i							
0	1	2	3	4	5	6	7	8	9
01	15	26	56	32	67	81	22	55	19



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[j]* então
- 4. troca *A[i] ↔ A[j]*

		i				j				
0	1	2	3	4	5	6	7	8	9	
01	15	26	56	32	67	81	22	55	19	5



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

		i								
0	1	2	3	4	5	6	7	8	9	
01	15	26	56	32	67	81	22	55	19	5



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

j											
0	1	2	3	4	5	6	7	8	9		
01	15	22	56	32	67	81	26	55	19		



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

								j	
0	1	2	3	4	5	6	7	8	9
01	15	22	56	32	67	81	26	55	19



- 1. Para $i \leftarrow 0$ até comprimento[A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[j]* então
- 4. troca *A[i] ↔ A[j]*

									j		
	0	1	2	3	4	5	6	7	8	9	
	01	15	22	56	32	67	81	26	55	19	5



- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca $A[i] \leftrightarrow A[j]$

									j	
0	1	2	3	4	5	6	7	8	9	
01	15	22	56	32	67	81	26	55	19	57



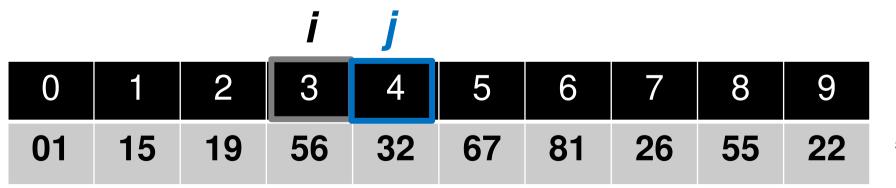
58

- 1. Para $i \leftarrow 0$ até comprimento [A] 1 faça
- 2. Para $j \leftarrow i+1$ até comprimento[A] faça
- 3. Se *A[i] > A[i]* então
- 4. troca *A[i]* ↔ *A[j]*

		i							j	
0	1	2	3	4	5	6	7	8	9	
01	15	19	56	32	67	81	26	55	22	į

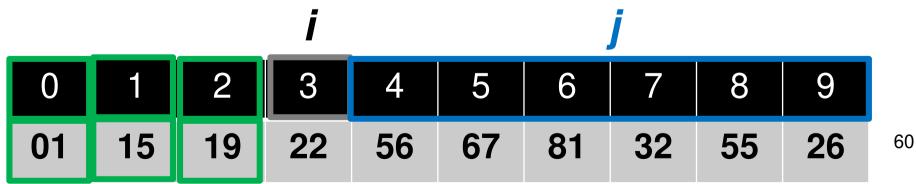


E assim por diante...



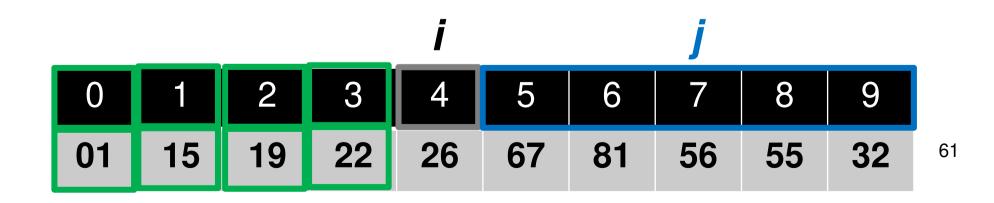


À esquerda: arranjo ordenado



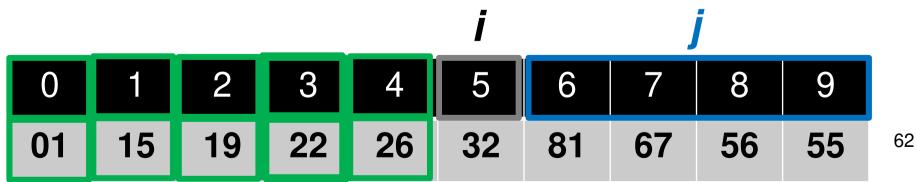


À esquerda: arranjo ordenado



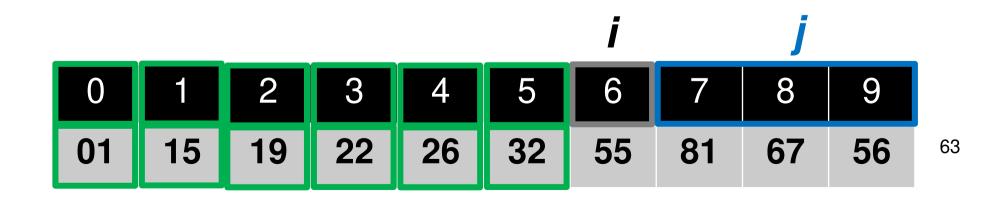


À esquerda: arranjo ordenado



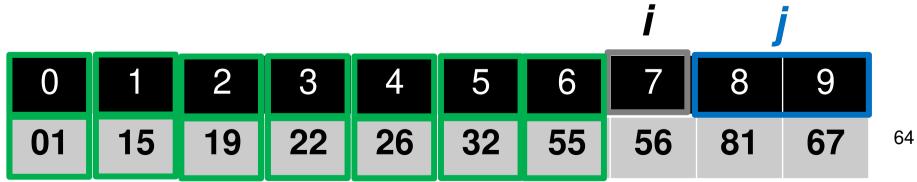


À esquerda: arranjo ordenado





À esquerda: arranjo ordenado





Arranjo ordenado!

i

0	1	2	3	4	5	6	7	8	9
01	15	19	22	26	32	55	56	67	81



Estrutura de Dados II

Prof. Me. Pietro M. de Oliveira