**Bubble Sort**

O **bubble sort** realiza múltiplas passagem por uma lista. Ele compara itens adjacentes e troca aqueles que estão fora de ordem. Cada passagem pela lista coloca o próximo maior valor na sua posição correta. Em essência, cada item se desloca como uma “bolha” para a posição à qual pertence.

https://panda.ime.usp.br/panda/static/pythonds\_pt/05-OrdenacaoBusca/OBubbleSort.html

**Shell Sort**

O **shell sort**, às vezes chamaado de “ordenação por incrementos diminutos”, melhora a ordenação por inserção ao quebrar a lista original em um número menor de sublistas, as quais são ordenadas usando a ordenação por inserção. A forma única como essas sublistas são escolhidas é a chave para o shell sort. Em vez de quebrar a lista em sublistas de itens contíguos, o shell sort usa um incremento i, às vezes chamado de **gap**, para criar uma sublista escolhendo todos os itens que estão afastados i itens uns dos outros.

https://panda.ime.usp.br/panda/static/pythonds\_pt/05-OrdenacaoBusca/OShellSort.html

**Sort Python**

O método sort() permite que você organize uma lista em ordem ascendente ou descendente. Ele recebe argumentos somente nomeados: key e reverse. reverse determina se a lista é ordenada em ordem ascendente ou descendente. key é uma função que gera um valor intermediário para cada elemento. Esse valor é usado para fazer as comparações durante o processo de ordenação.