

3. Procedure bubble ($T[1..n]$)

```
for i ← 1 to n do
  for j ← 1 to n-i do
    if  $T[j] > T[j+1]$  then
      temp ←  $T[j]$ 
       $T[j] \leftarrow T[j+1]$ 
       $T[j+1] \leftarrow temp$ 
```

Idea: po i -tej iteracji i największych elementów tablicy jest ustawionych na prawidłowej pozycji. W $i+1$ iteracji ponownie sortujemy elementy i przesuwamy większego na prawo, jeśli są nieporządkowane.

Złożoność czasu:

Bubble - wewnętrzna pętla wykonuje się n razy dla $i = 0, n-1$ razy dla $i = 1, n-2$ razy dla $i = 1, \dots, 1$ raz dla $i = n-1$, zatem łączną wykonuje się $\underbrace{n + n-1 + n-2 + \dots + 2 + 1}_{\text{suma ciągu arytm.}} = \frac{n+1}{2} \cdot n = \frac{n(n+1)}{2}$ razy.

Układa iteracji pętli wewnętrznej wykonujemy stałą liczbę operacji między RAM (przesuwanie i porównywanie elementów tablicy), zatem jego złożoność asymptotycznie to $O(n^2)$, podobnie jak Select. Insert optymalizujemy w przypadku listy (złożoność $O(n)$), w przypadku tablicy $O(n^2)$.

Złożoność pamięciowa: Algorytm sortuje tablicę w miejscu, zatem wystarczy nam n jednostek pamięci na jej przechowanie i jedna niezmieniona temp. Jego złożoność asymptotycznie to $O(n)$, podobnie jak dla algorytmów Insert i Select.

Realizacja danych: W przypadku danych przed wprowadzonymi najlepiej wybrać Insert ze złożonością Inserta, Bubble w takim przypadku wymaga jedynie przesuwania elementów tablicy. Bubble, podobnie jak Select, nie wymaga obsługi danych ma złożoność $O(n^2)$.

Wielkość rekurencji: Bubble dla dużych wartości danych może być bardzo kosztowny, szczególnie dla posortowanych danych (co by wykonało najwięcej przesunięć - $O(n^2)$), podobnie jak Insert w tym przypadku ustępuje algorytmowi Select.

Stabilność: Bubble jest stabilny - dla elementów o tej samej wartości nie zamienia ich miejscami. Jest pod tym względem podobny do Insert, gdzie Select może być stabilny.

Intensywność: Można zauważyć, że możliwe jest optymalizacja Bubble sortu. Wystarczy sprawdzić, czy wykonujemy za dużo przesunięć - jeśli w danej iteracji pętli wewnętrznej ich nie było, wtedy tablica już jest posortowana i możemy zakończyć algorytm.