Csi503-s21-lecture02

Thursday, February 4, 2021

input: $\langle 0, -5, 10, 5, 0 \rangle$
output: $\langle -5, 0, 0, 5, 10 \rangle$

Problem: sort in increasing order

$\langle a_1, a_2, a_3, a_4, a_5 \rangle$

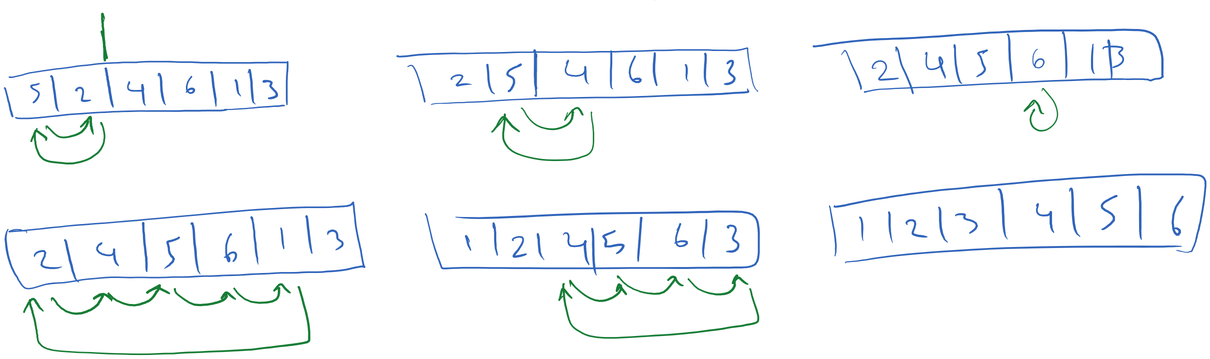
$a_1 < \underbrace{a_2 < a_3}_{\times} < a_4 < a_5$

---

pseudocode

insertion sort

```
Insertion-sort(A, n)                          cost    times
  for j = 2 to n                              c1      n
      Key = A[j]                              c2      n-1
      i = j - 1                               c4      n-1
      while i > 0 and A[i] > Key              c5      sum_{j=2}^{n} t_j
           A[i+1] = A[i]                      c6   }  sum_{j=2}^{n} (t_j - 1)
           i = i - 1                          c7   }
      A[i+1] = key                            c8      n-1
```



Correctness

Induction | **Correctness based on invariants**

① base case $P(n = n_0)$ ⟷ (initialization) invariant is true before you start

② inductive step $P(K) \longrightarrow P(K+1)$ ⟷ (maintenance) invariant holds from iteration to iteration

(termination) when loop ends we have the expected output proof of correctness for insertion sort

initialization: $A[1]$ is sorted

maintenance: will shift $A[j-1], A[j-2], \ldots$ to right until finding correct sorted place for key

termination: outer loop terminates when $j = n+1$. At that point, everything before $n+1$ is already sorted based on the invariant

---

Complexity Analysis

characterize based on input size

- e.g.: graphs $G\langle V, E\rangle \longrightarrow$ input $\underset{n}{|V|}, \underset{m}{|E|}$

insertion-sort:

best case: $t_j = 1 \longrightarrow$ cost: $an + b$

worst case: $t_j = j \longrightarrow$ cost: $an^2 + bn + c$

average case: ? $t_j = j/2 \longrightarrow$ cost: $a'n^2 + b'n + c$

$$f(n) = \Theta(n^2)$$