1.Having specified the insertion sort algorithm, we then **argue** that it correctly sorts, and we analyse its **running time**.

2.**notation**

3.**divide-and-conquer** approach

4.**take as a parameter** an array A[1...n] containing a sequence of length n that is to be sorted.

5.sort the input numbers **in place**

6.Array indices **appear above** the rectangles.

7.**shaded**/black rectangles

8.**arrows**

9.loop invariants

10.the **pile** of cards

11.Initialization-Maintenance-Termination(**properties**)

12.along with

13.cause the **loop** to **terminate**

14.**best-case/worst-case/average-case** running time

15.**constant-time** instruction/operation

16.provide a **guarantee** that

17.**on average**

18.**yield** an expected running time

19.**order/rate of growth**

20.**incremental** approach

21.**bottom out**(terminate the recursion)

22.**motif**

23.**prior to**

24.**recurrence (equation)**

25.straightforward