[HW#1.P5]
Insertion sort가 correct함을 loop-invariant를 이용해서 증명

<Loop invariant>
A[1…j-1]까지 정렬한다.

Initialization: 루프의 반복이 시작되기전 j=2일 때 루프불변성이 true인지 확인한다. 이 때 j=2에 대해서 A[1...j-1]은 A[1]한개의 원소로 구성되어있고 하나의 원소만 있으므로 정렬되어있다. 따라서 루프의 첫 반복이 시작되기전 루프불변성의 초기조건은 성립된다.

Maintenance: 유지조건에서 매 반복할 때마다 루프 불변성이 유지되는지 확인한다. 루프의 while문은 A[j]가 될 때까지 A[j-1], A[j-2], A[j-3]...는 A[j]값이 나올 때까지 계속 루프가 돌게 되고, 이 루프가 반복되면서 A[1...j]는 동일한 원소들을 정렬한 상태로 갖게 된다.

Termination: 종료조건에선 j가 배열의 length보다 커질 때까지 1씩 증가하다가 A[j]가 되었을 때 종료된다. 루프의 유지조건에서 A[j]가 되게되면, 그 앞에 있는 모든 원소들까지 정렬되어있으므로, 이는 알고리즘의 타당함을 의미한다.

==

[HW#1.P6]
Merge sort 중 merge invariant를 이용해서 증명
함 수 가 correct 함 을 loop- invariant를 이용해서 증명

<Loop invariant>
루프를 반복함으로써 A[p...k-1]은 L[1...n1+1]과 R[1...n2+1]의 원소 중 가장 작은 값을 가진 k-p개 원소를 정렬된 순서로 가진다. 또한 L[i]와 R[j]는 각 배열에서 A로 복사되지 않은 원소 중 가장 작은 값을 가지게 된다.

Initialization: 루프의 반복이 시작되기전에는 k=p이므로 A[p...k-1]은 빈 배열이 된다.. 그리고 초기조건인 i=j=1은 L[i]와 R[j]각 각 원소를 1개씩만 가지게 되어 각각 A로 복사되지 않은 가장 작은 원소를 가지게 된다

Maintenance: 유지조건에서 매 반복할 때마다 루프 불변성이 유지되는지 확인한다. 먼저 L[i] <= R[j]의 경우 L[i]는 A로 복사되지 않은 가장 작은 원소이다. A[p...k-1]은 k-p개의 가장 작은 원소를 가지고 있으므로, L[i]를 A[k]로 복사하면(A[k]=L[i]) 부분 배열 A[p...k]는 k-p+1개의 가장 작은 원소를 정렬된 순서로 저장하게 된다. k와 i를 증가시키는 것은 다음 반복에서의 루프 불변성의 유지조건을 만들어준다. L[i]>R[j]인경우에도 동일한 유지조건이 확립한다.

Termination: 종료조건은 k=r+1인 경우이다. 루프 불변성에 의해 부분 배열 A[p...k-1], 즉 A[p...r]은 L[1...n1+1]과 R[1...n2+1]에서 가장 작은 r-p+1개의 원소를 정렬된 순서로 저장한다. 배열 L과 R은 모두 합쳐 n1+n2+2=r-p+3개의 원소를 가지게 된다. 가장 큰 두 개의 원소를 제외하고 모두 A로 복사되는데, 남은 두 원소는 경계카드(sentinel)이다.

==

[HW#1.P7]
2-4 역위

a.
(0,4),(1,4),(2,4), (3,4), (3,4)

b.
내림차순으로 정렬된 경우가 가장 많은 inversion을 갖는 경우이다. 즉 n(n-1)/2 개이다.

c.

d.

==

[HW#1.P8]
2.3-7

먼저 n개 정수의 집합 S를 sorting한다.
여기선 병합정렬을 이용해 sorting할 경우 시간복잡도는 n이된다.(위의 문제 참조)
그리고 이후 두 원소의 합이 x가 되는 경우를 확인하기 위해 S[i]-x를 만족하는 y가 S의집합에 있는지 binary Search로 검색한다.(nlogn)
Binary Search의 경우 O(logn)의 시간복잡도를 가진다.
이를 곱하면 O(nlogn)*O(logn) = O(nlogn^2)의 시간복잡도를 가지게 된다.

==

[HW#1.P9]
2.3-3

* Basis step
$n=2$인 경우
$T(2)=2$이고
해는 주어진 조건에 따라 역시 $2\log 2$로
같은 값을 가지므로 True 이다.

* Inductive step
귀납가정에 의해 2^k가 True인 경우
2^{k+1}도 True가 되는지 확인해 본다
$n=2^{k+1}$

$$T(2^{k+1}) = 2T\left(\frac{2^{k+1}}{2}\right)+2^{k+1}$$
$$= 2T(2^k)+2^{k+1}$$
$$= 2(2^k\log 2^k)+2^{k+1}$$
$$= 2^{k+1}(\log 2^k+1)$$
$$= 2^{k+1}\log 2^{k+1}$$

$\therefore$ $T(2^{k+1})$인 경우에도
nlogn의 해를 가짐을
확인하였으므로 점화식의 해는
True 이다.

==

[HW#1.P10]

2.3-3

$\boxed{\text{Solution 1}}$

$$
\begin{aligned}
T(n) &= T(n-1)+n \\
&= T(n-2)+(n-1)+n \\
&= T(n-3)+(n-2)+(n-1)+n \\
&= T(n-4)+(n-3)+(n-2)+(n-1)+n \\
&\;\;\vdots \\
&= T(0)+1+2+3+\cdots+(n-2)+(n-1)+n \\
&= T(0)+\frac{n(n+1)}{2} \\
&= \Theta(1)+\frac{n^2+n}{2} \Rightarrow \Theta(n^2)
\end{aligned}
$$

$\boxed{\text{Solution 2}}$

$n>1$ $c>1$에 대하여 다음의 공식이 만족된다.

$$
\begin{aligned}
T(n) &= T(n-1)+n \\
&\leq c(n-1)^2+n \\
&\leq c(n^2-2n+1)+n \\
&\leq cn^2-2cn+c+n \\
&\leq cn^2-(2c-1)n+c \\
&\leq cn^2
\end{aligned}
$$

$\therefore\ O(n^2)$