Lemma: Binomial Heap insertion sequence of n elements takes $\lambda(n)$ time

Proof: Idea ⟹ For every new insertion, #comparisons = # position of Least Significant Zero (LSZ) + 1

(Position of = cost − 1)

n (input size increasing)

Bit Position: 4	3	2	1	0	Position of least significant zero (LSZ)
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	0
0	0	0	1	1	2
0	0	1	0	0	0
0	0	1	0	1	1
0	0	1	1	0	0
0	0	1	1	1	3
0	1	0	0	0	0
0	1	0	0	1	1
0	1	0	1	0	0
0	1	0	1	1	2
0	1	1	0	0	0
0	1	1	0	1	1
0	1	1	1	0	0
0	1	1	1	1	4

Total $\approx$ 15.

Cost $\propto$ 2(15)

LSZ cases:

don't cares	takes time
....0 ⟶	1
....01 ⟶	2
...011 ⟶	3
..0111 ⟶	4
⋮	⋮
$0\underbrace{111\ldots1}_{K-1}$ ⟶	K time

⟹ Total (amortized) cost of n successive inserts

$$= \frac{n}{2}\times 1 + \frac{n}{4}\times 2 + \frac{n}{8}3 + \cdots + \frac{n}{2^k}\cdot k$$

$$= \sum_{i=1}^{k} \frac{n}{2^i}\times i, \quad \text{where } k = \lg n$$

$$< \sum_{i=0}^{\infty} \frac{n}{2^i}\times i$$

$$= n\sum_{i=1}^{\infty}\frac{i}{2^i} = 2n$$

⟹ Total amortized cost of n successive inserts

$$\leq 2n$$
$$= O(n)$$

⟹ Amortized cost of each insert operation = O(1)