# Inducción Estructural HeapSort

## Paso Base:

Tenemos una lista de un solo elemento, al ser la lista de un solo elemento el heap ya es válido al estar ordenado. Por lo tanto esto quiere decir que este paso base se cumple.

## Hipótesis Inductiva:

Se asume que el algoritmo heapsort funciona para cualquier lista de tamaño n

## Paso inductivo:

Lo que se hace primero es convertir la lista en un maxheap, esto se hace con la operación heapificación desde el último nodo padre hasta la raíz asegurándose de esta manera que se cumpla la propiedad de que cada nodo padre es mayor o igual que sus hijos.

Supongamos que al momento de añadir n+1 el algoritmo asegura que la heapificación se mantenga válida después de la inserción. Sabemos por la hipótesis que cualquier lista n ya se ha convertido en un heap, por lo que la operación (Heapificación) se mantiene válida y hace que el heap completo de n+1 sea correcto.

Ahora vamos con sacar el elemento máximo, que es intercambiar la raíz con el último elemento de la lista, por lo que ahora el heap es de n y se aplica otra vez la operación para restablecer el max heap y como sabemos por la hipótesis, esta se reordena correctamente. Y así con el resto del heap.

Como la inducción es válida y el paso inductivo igual (El algoritmo funciona para una lista n+1 y asumimos que para la lista n funciona) asumimos que el algoritmo heapsort funciona para cualquier lista de tamaño n.