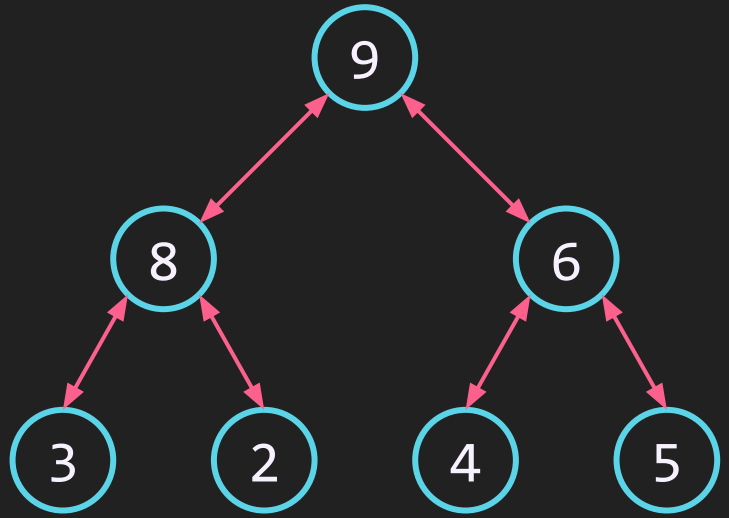
Taller 1
Binary Heaps

With 💜 by @vichoeq

Heap como árbol

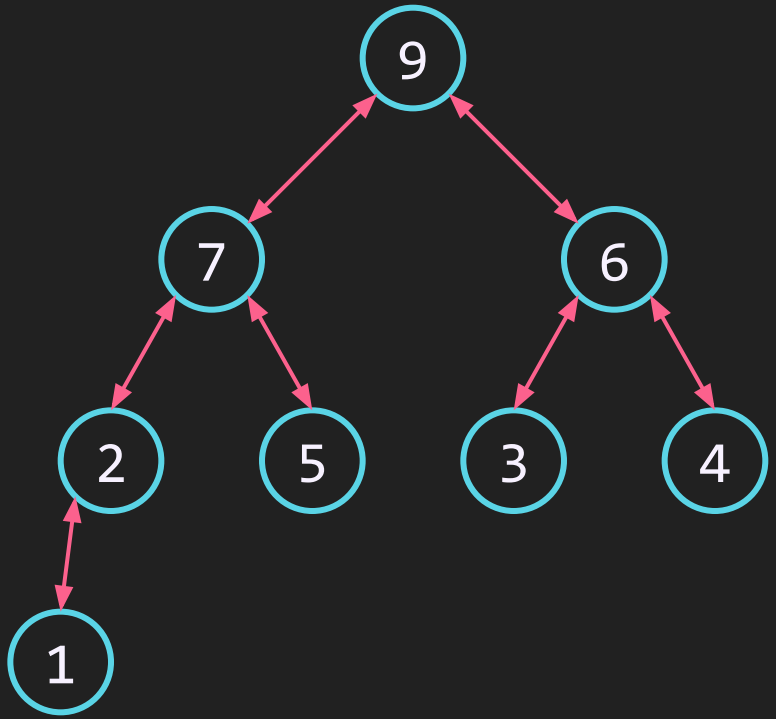
```c
struct heap;
typedef struct heap Heap;

struct heap
{
  Heap* left_child;
  Heap* right_child;
  Heap* parent;
  int value;
  int count;
  int height;
};
```

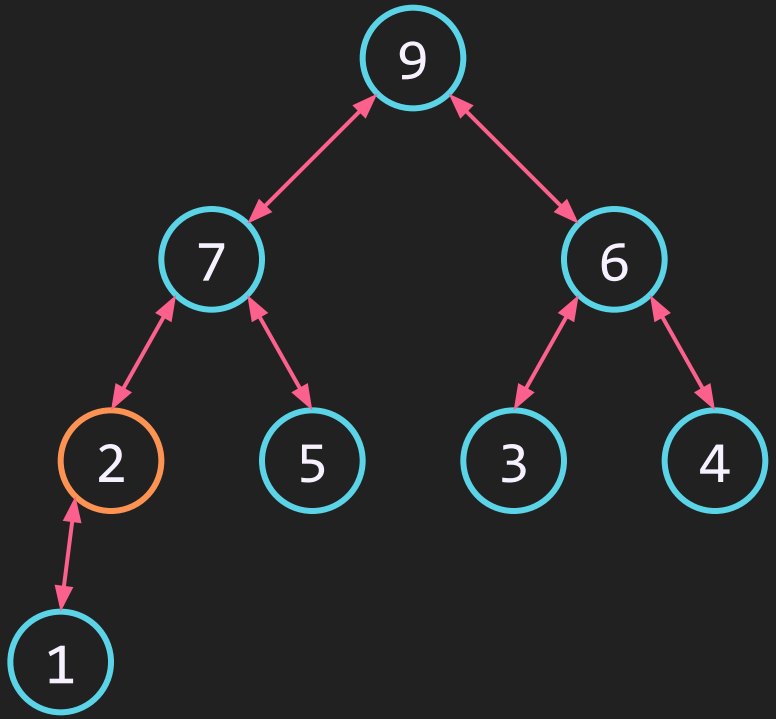


heap_insert

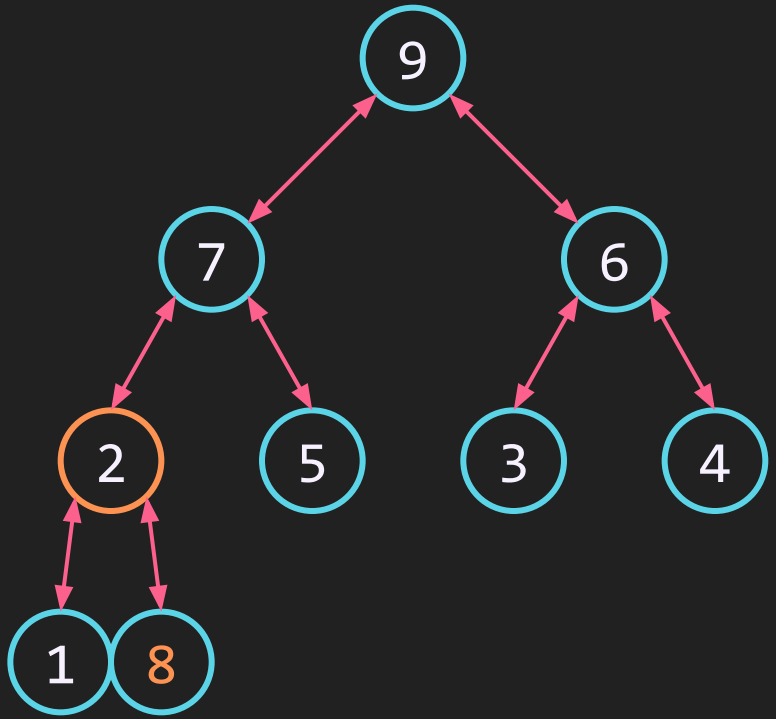
heap_insert(9, 8)



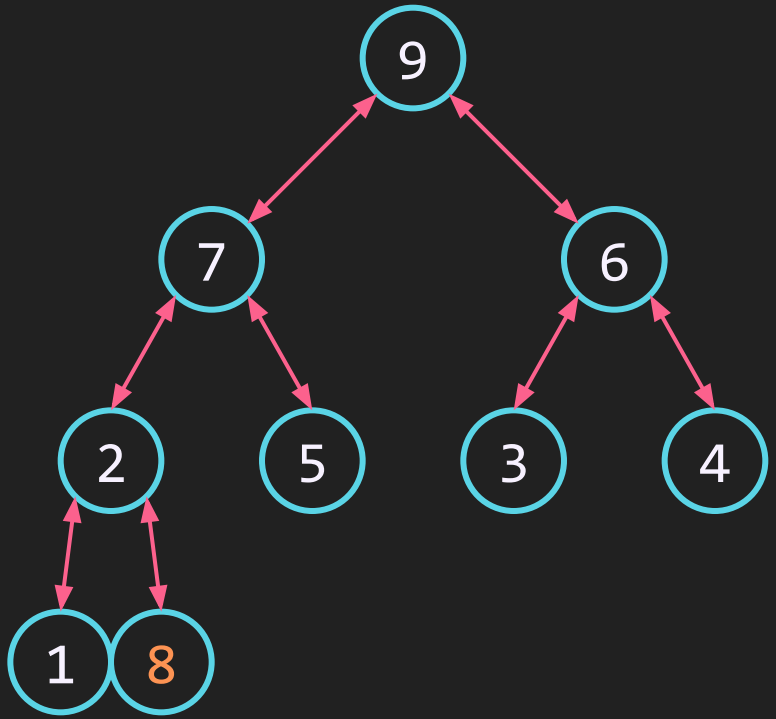
2 = insertion_parent(9)



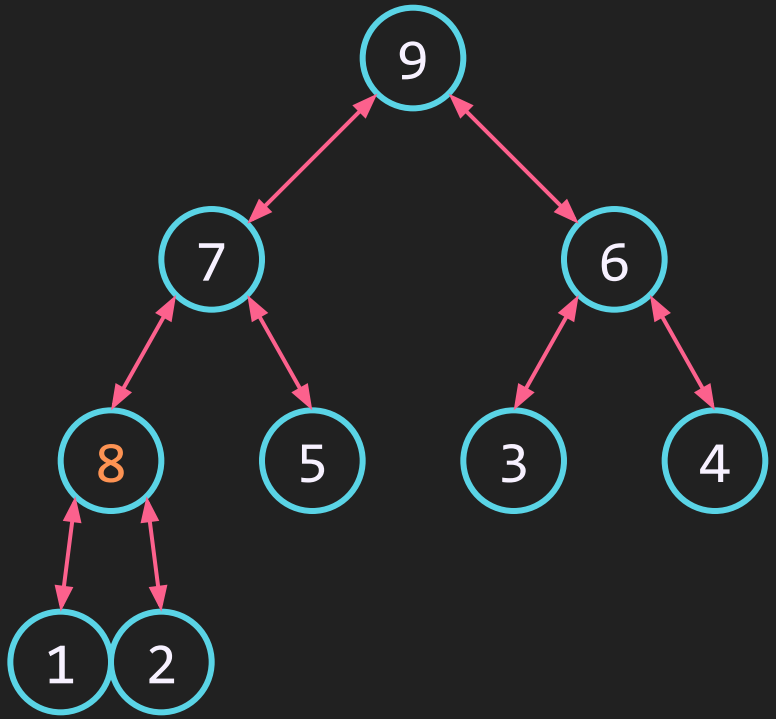
2.right_son = 8



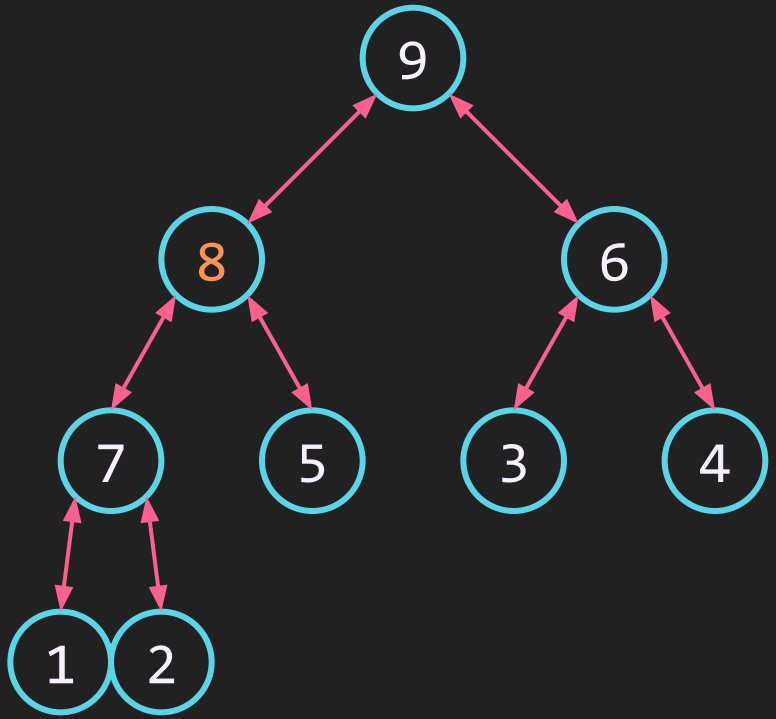
sift_up(8)



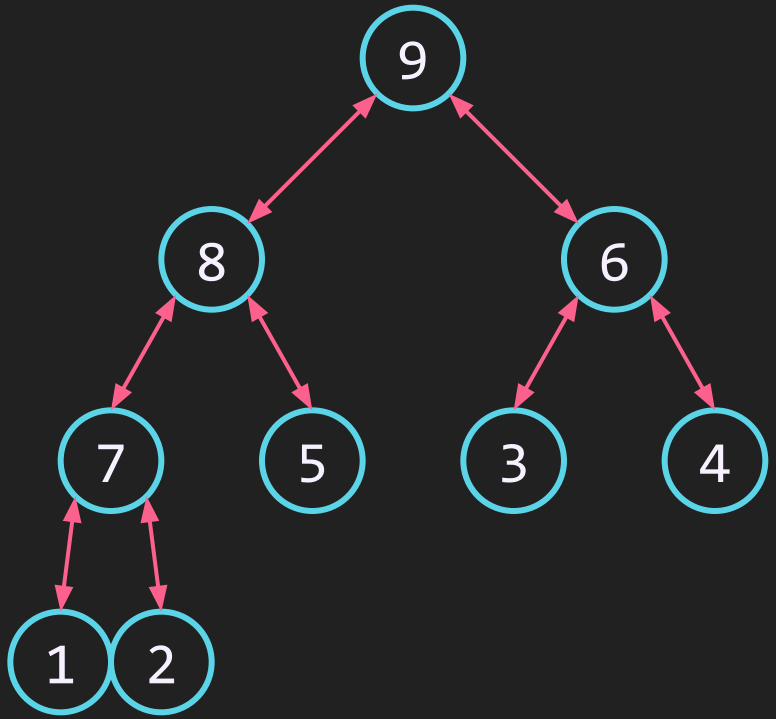
sift_up(8)



sift_up(8)

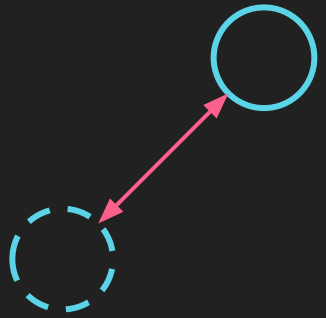


done

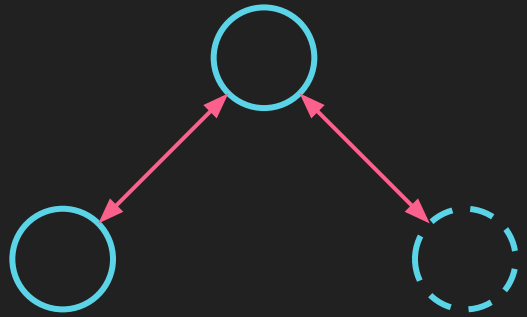


heap_insert - inserción por niveles

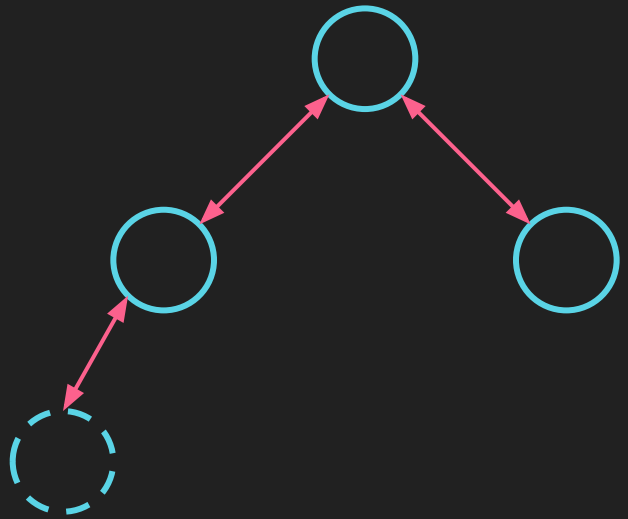
heap_insert - inserción por niveles



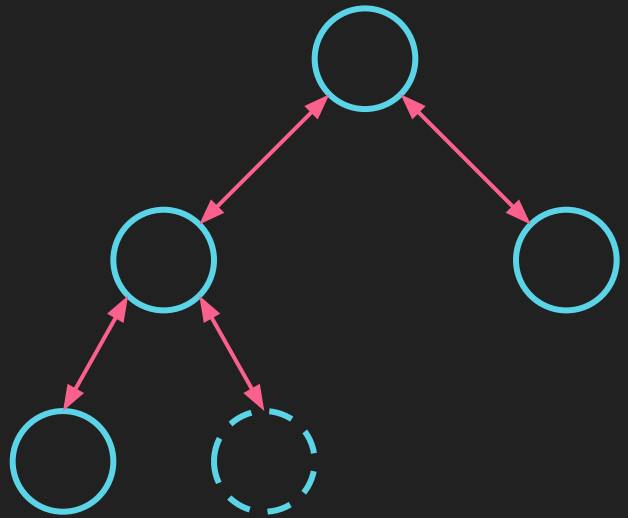
heap_insert - inserción por niveles



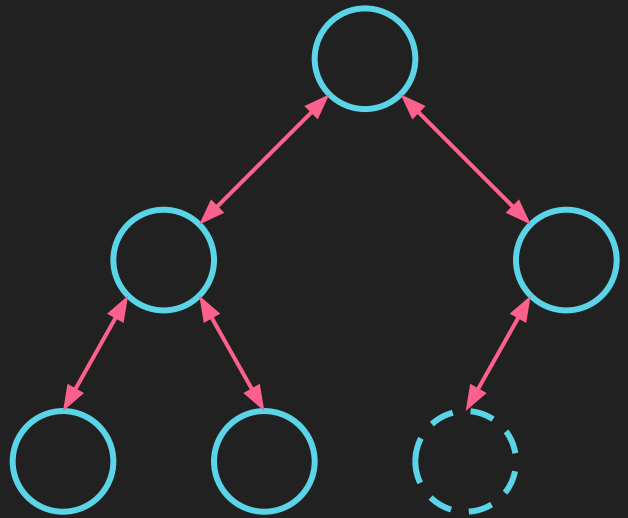
heap_insert - inserción por niveles



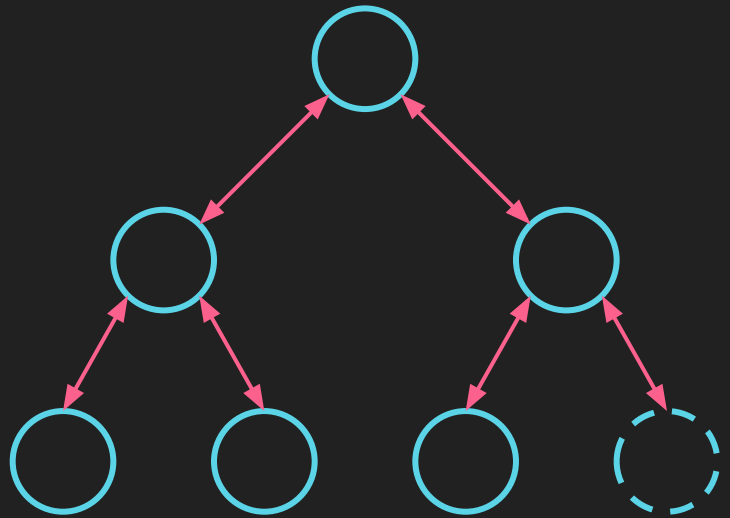
heap_insert - inserción por niveles



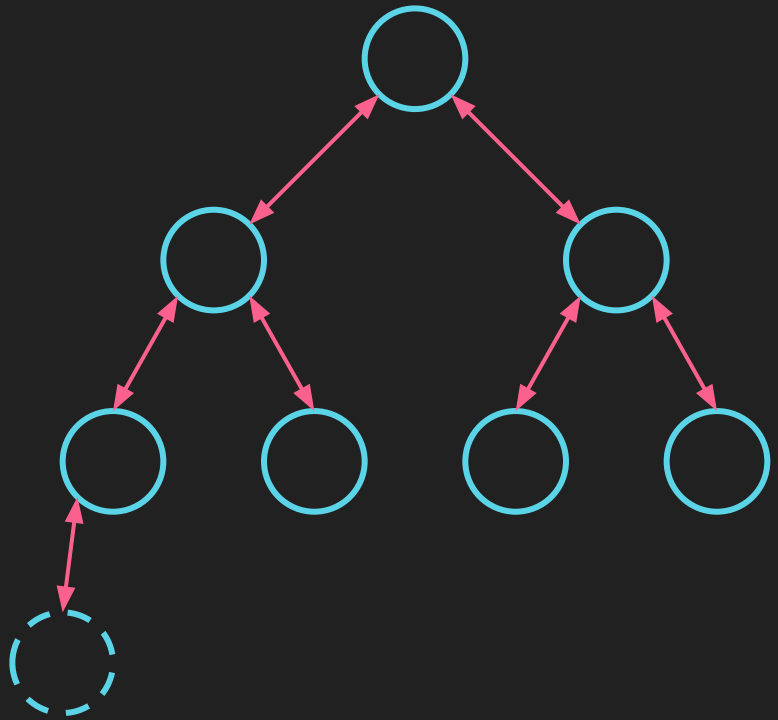
heap_insert - inserción por niveles



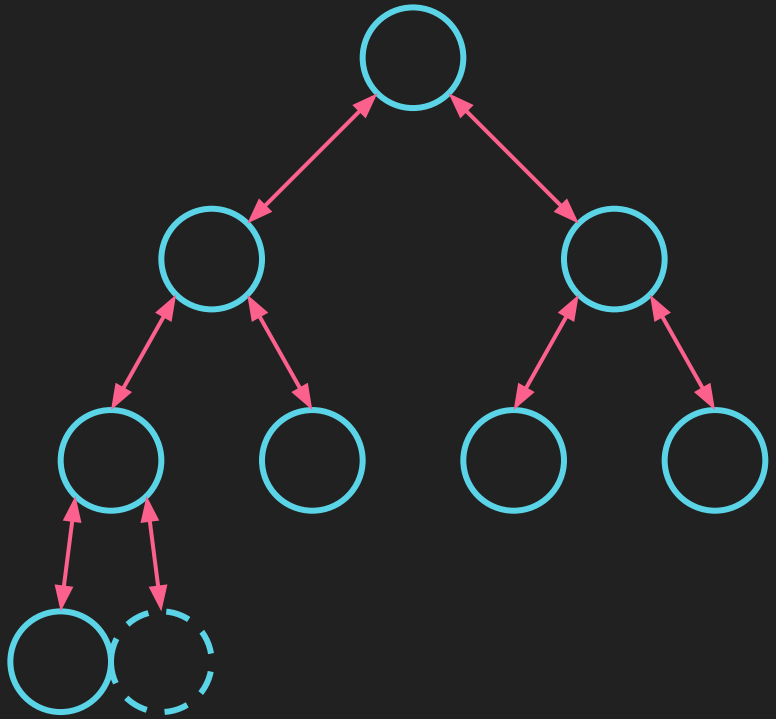
heap_insert - inserción por niveles



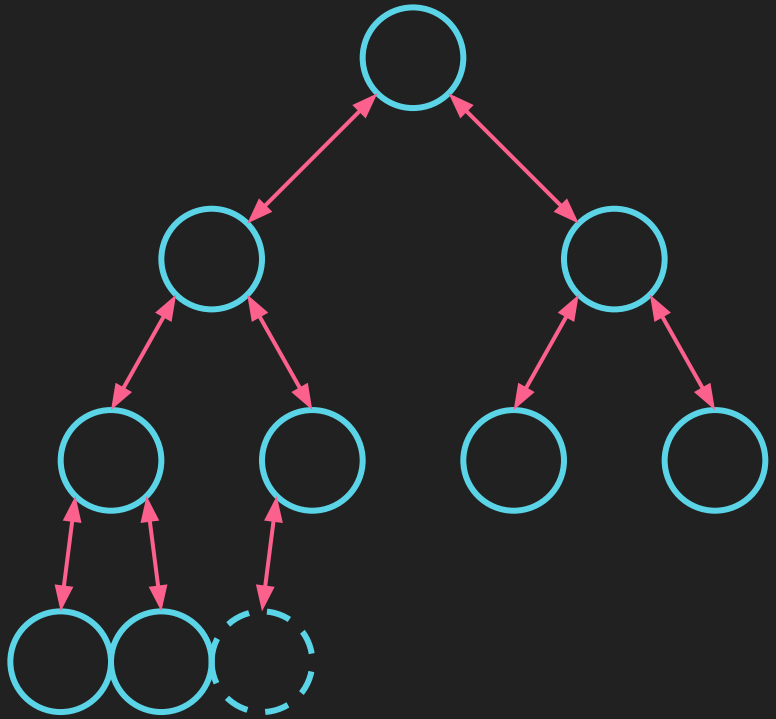
heap_insert - inserción por niveles



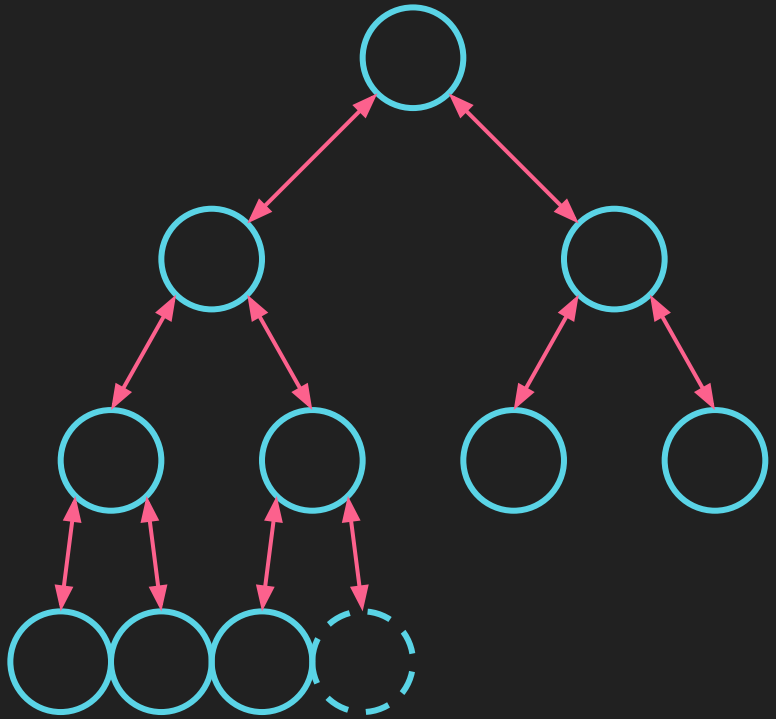
heap_insert - inserción por niveles



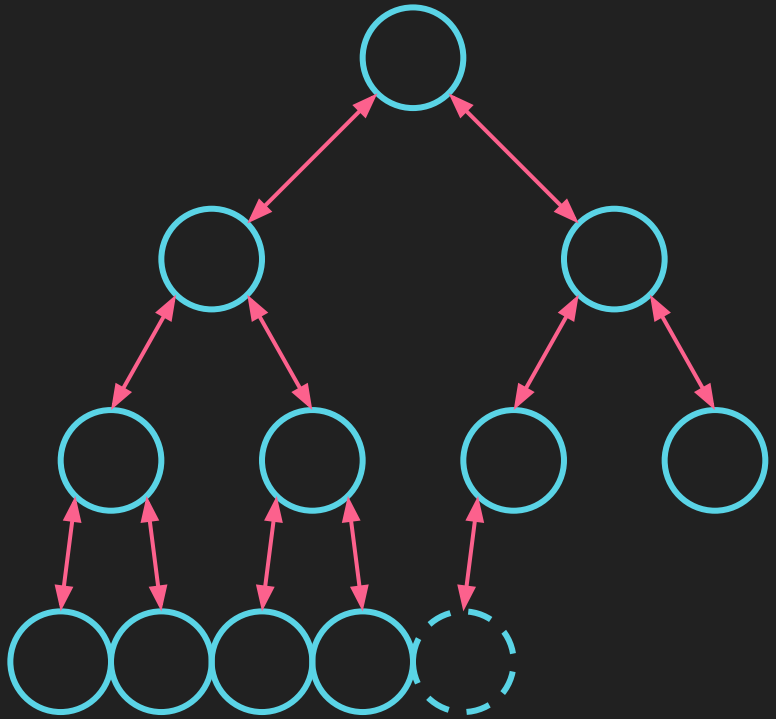
heap_insert - inserción por niveles



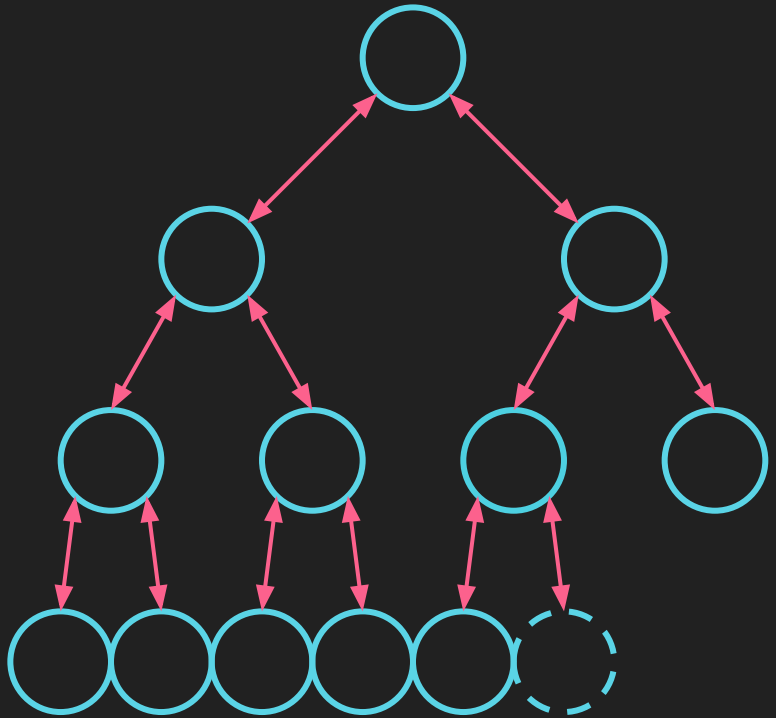
heap_insert - inserción por niveles



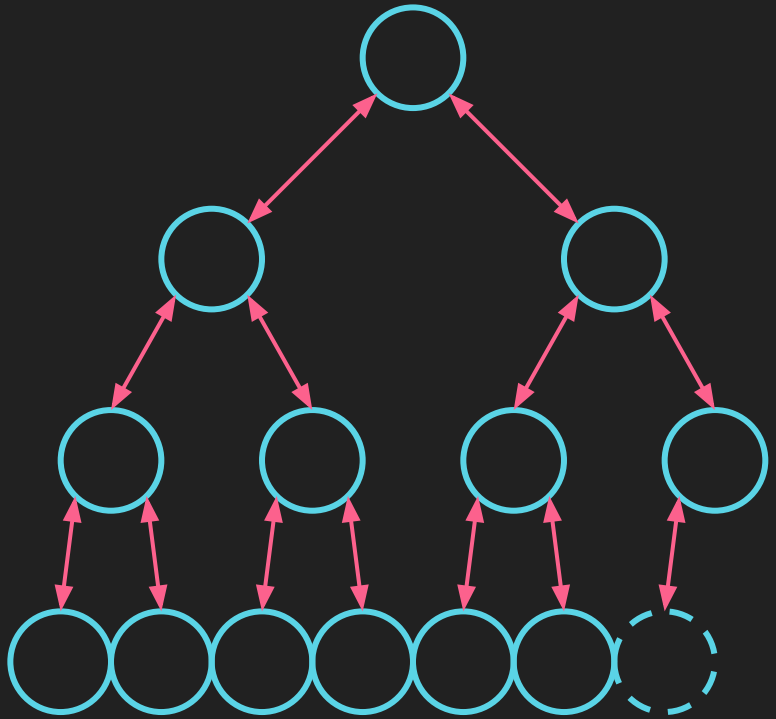
heap_insert - inserción por niveles



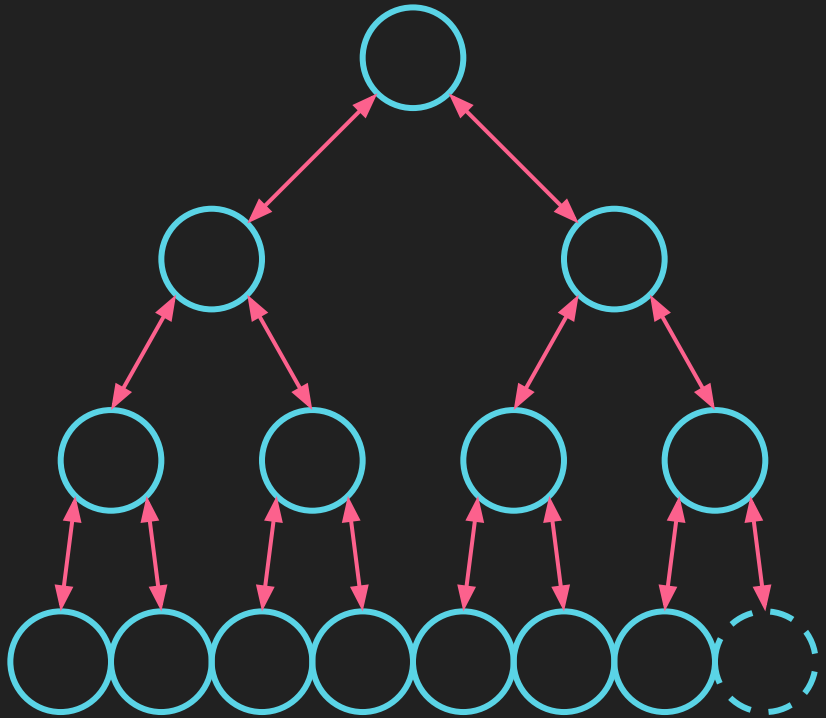
heap_insert - inserción por niveles



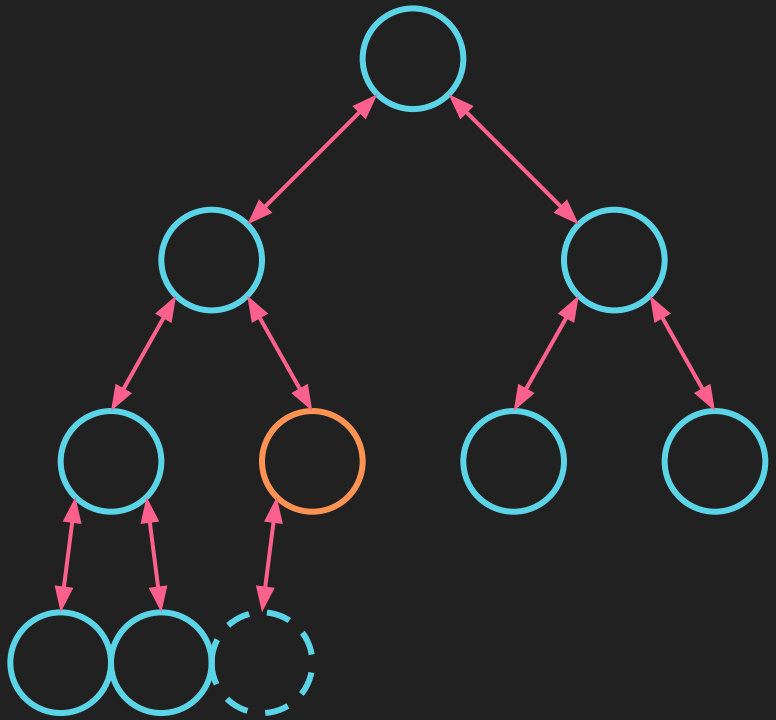
heap_insert - inserción por niveles



heap_insert - inserción por niveles



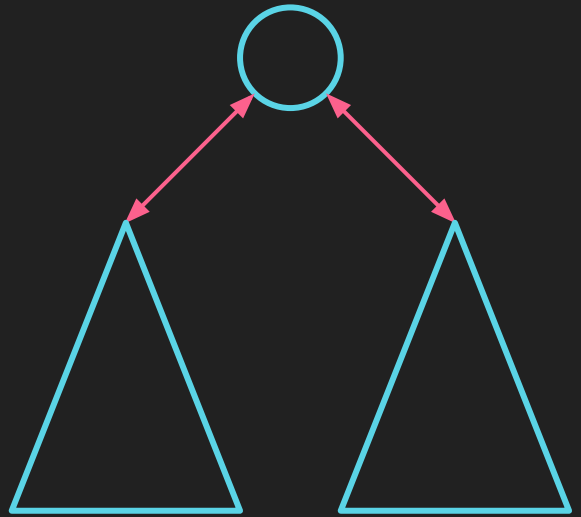
`insertion_parent` - padre del insertado



insertion_parent - padre del insertado

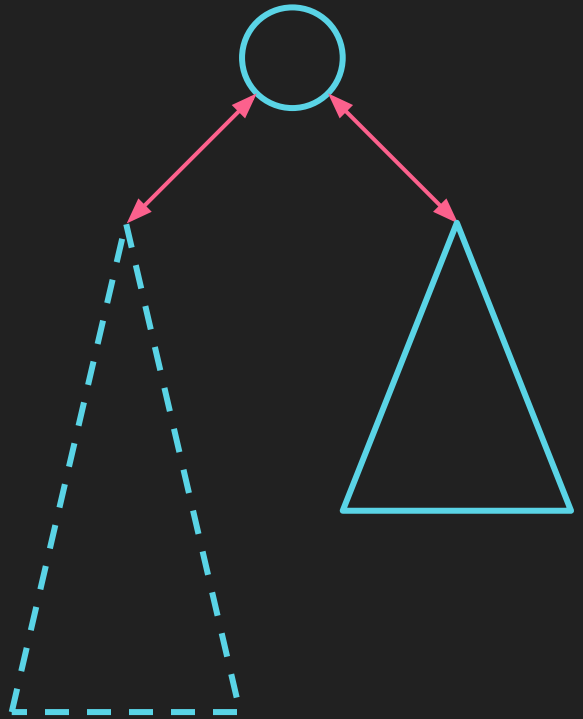
Caso 1: Heap no tiene 2 hijos

insertion_parent - padre del insertado



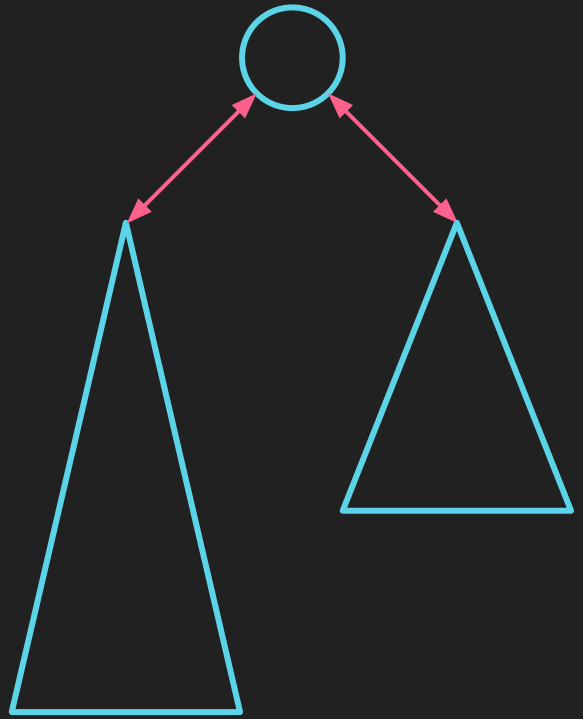
Caso 2: Ambos hijos están completos y son de la misma altura

insertion_parent - padre del insertado



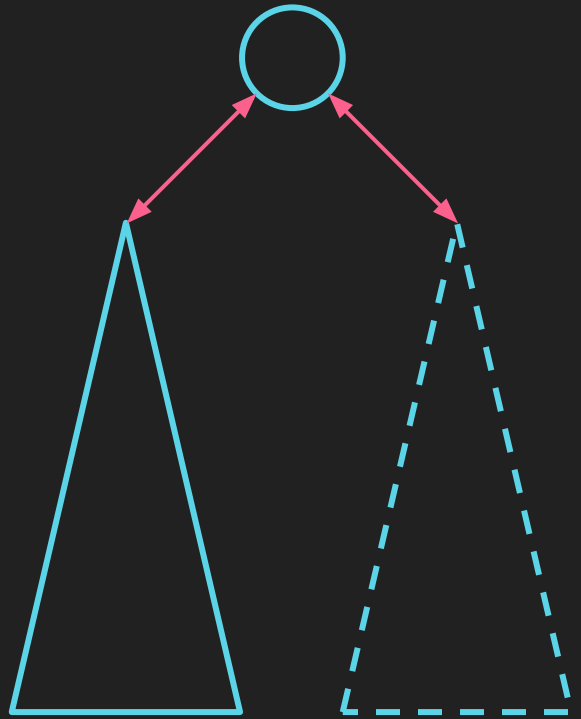
Caso 3: El hijo izquierdo está incompleto

insertion_parent - padre del insertado



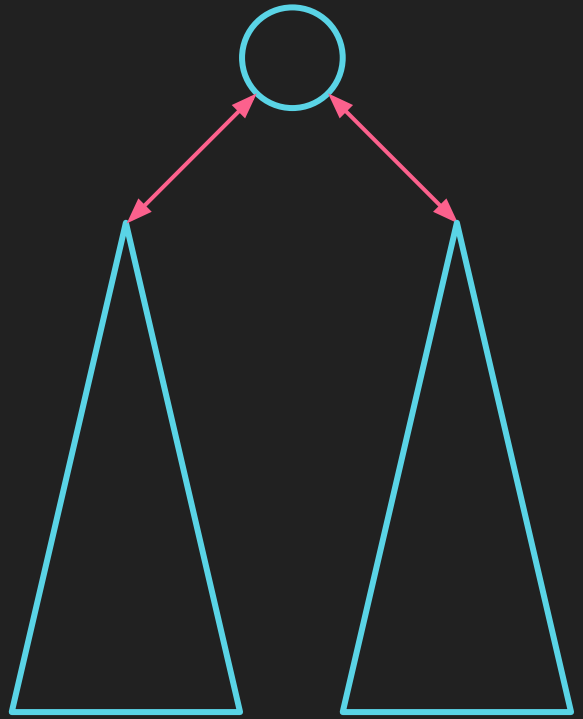
Caso 4: Ambos hijos están completos pero son de distinta altura

insertion_parent - padre del insertado



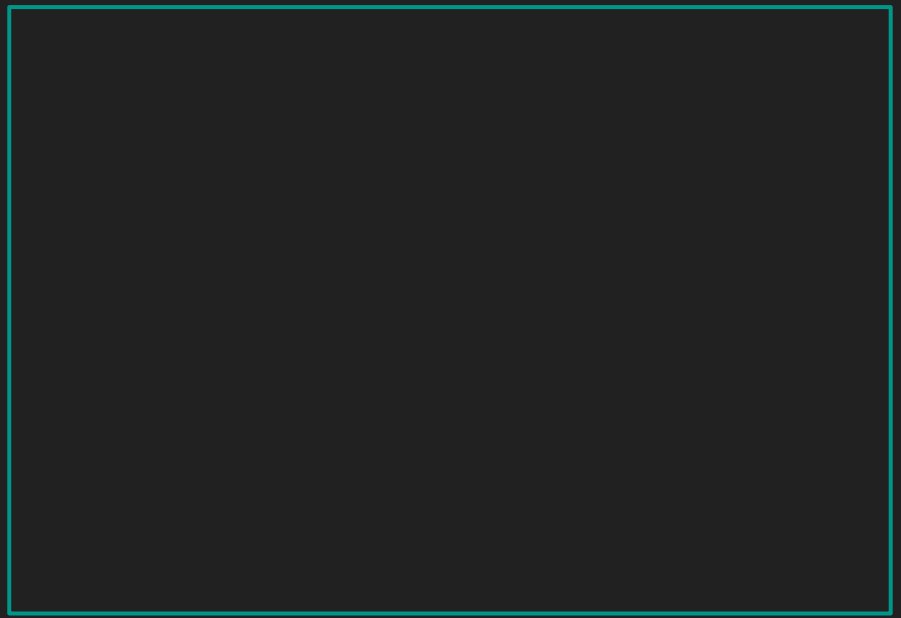
Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado

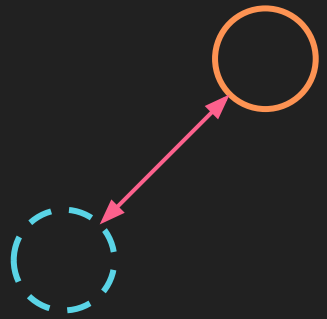


Caso 2: Ambos hijos están completos y son de la misma altura

`insertion_parent` - padre del insertado



insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

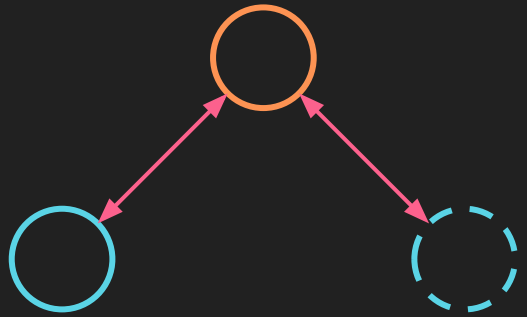
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

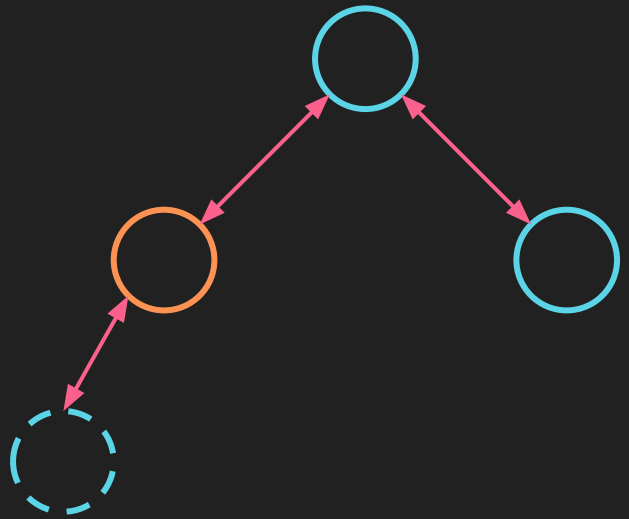
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

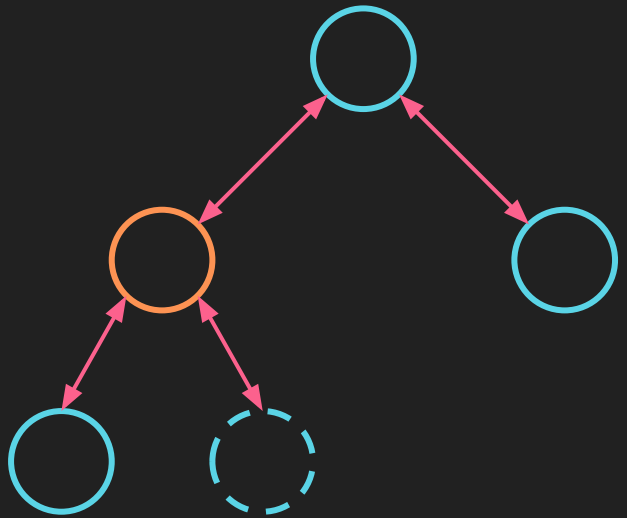
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

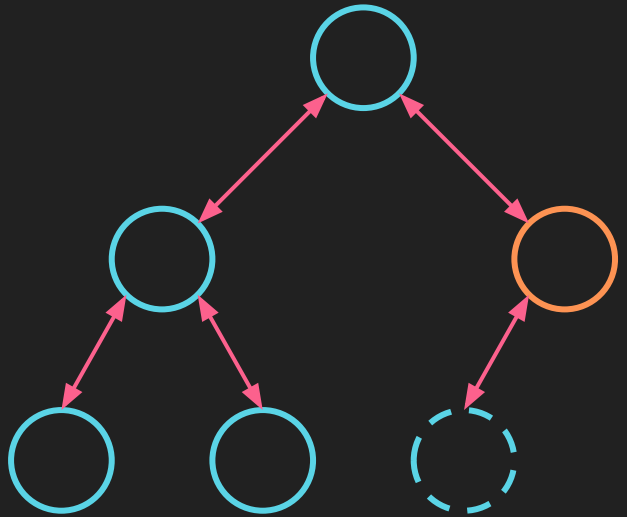
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

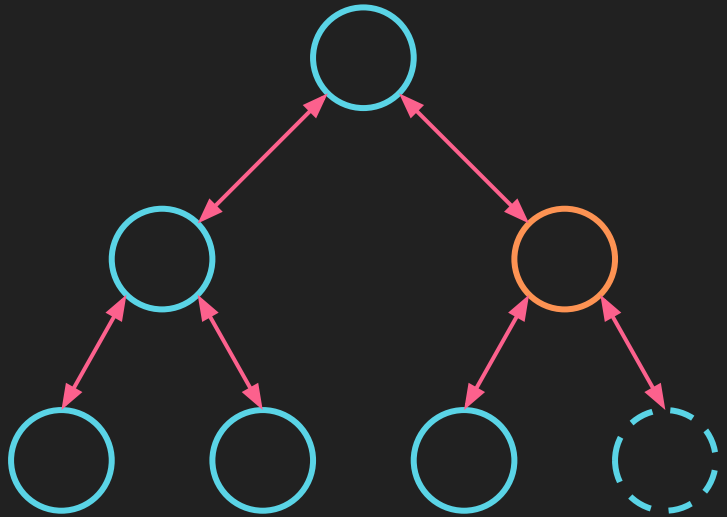
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

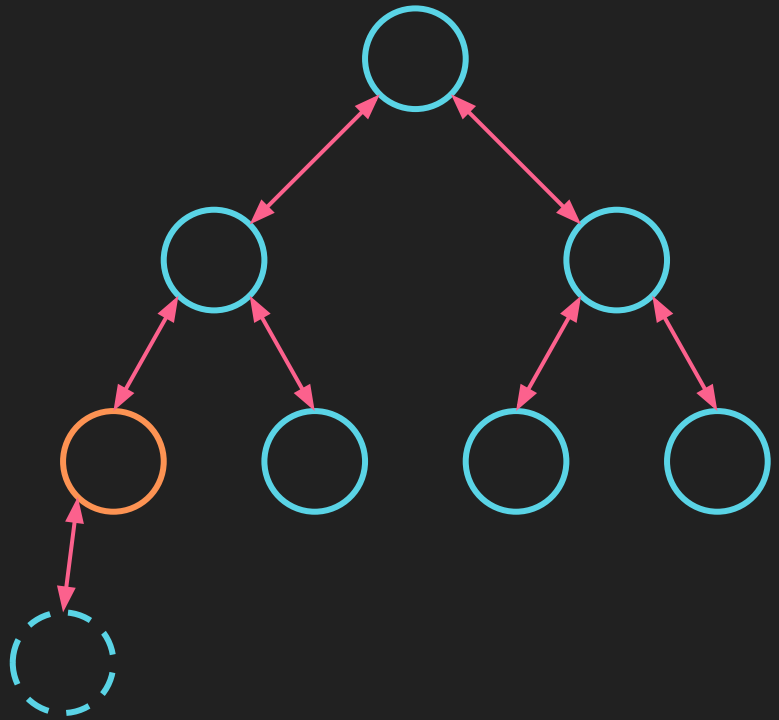
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

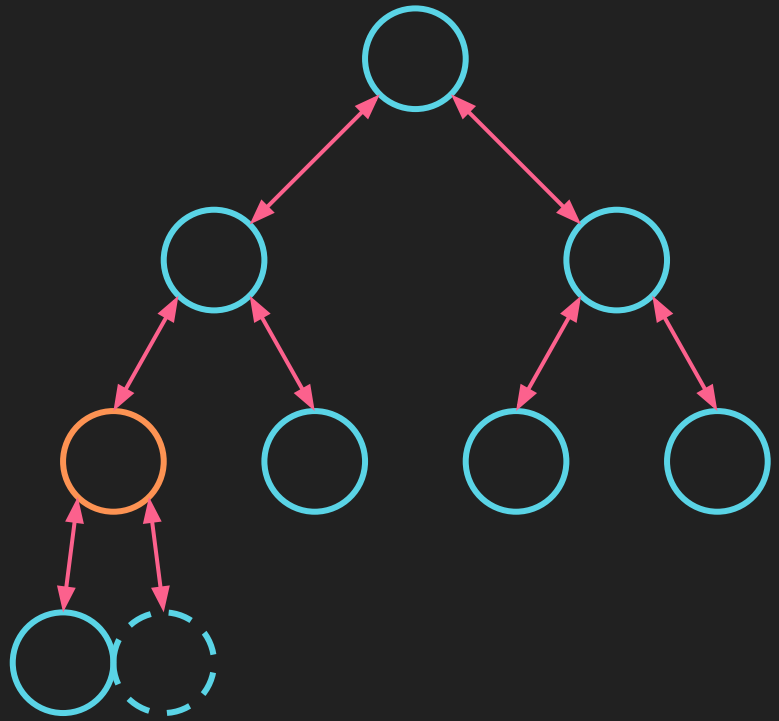
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

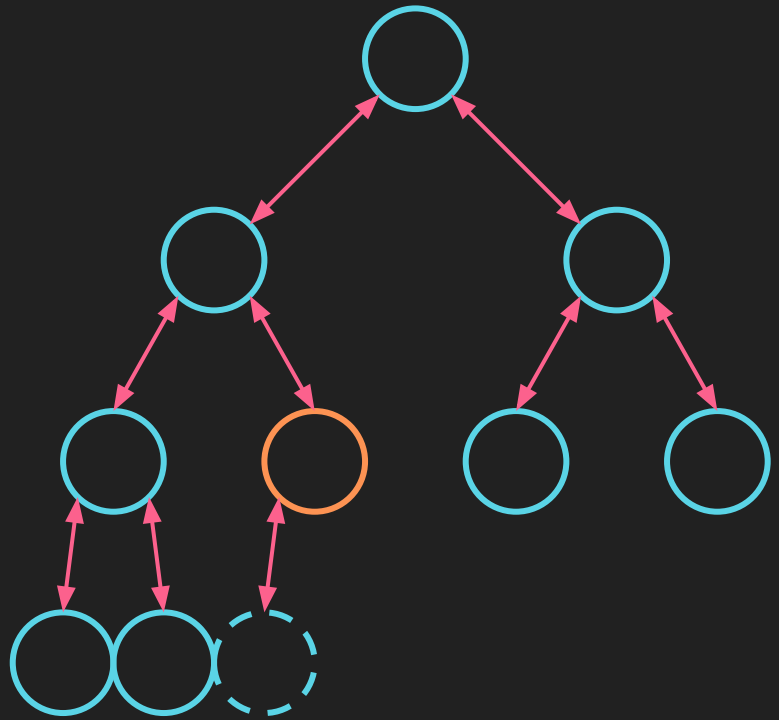
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

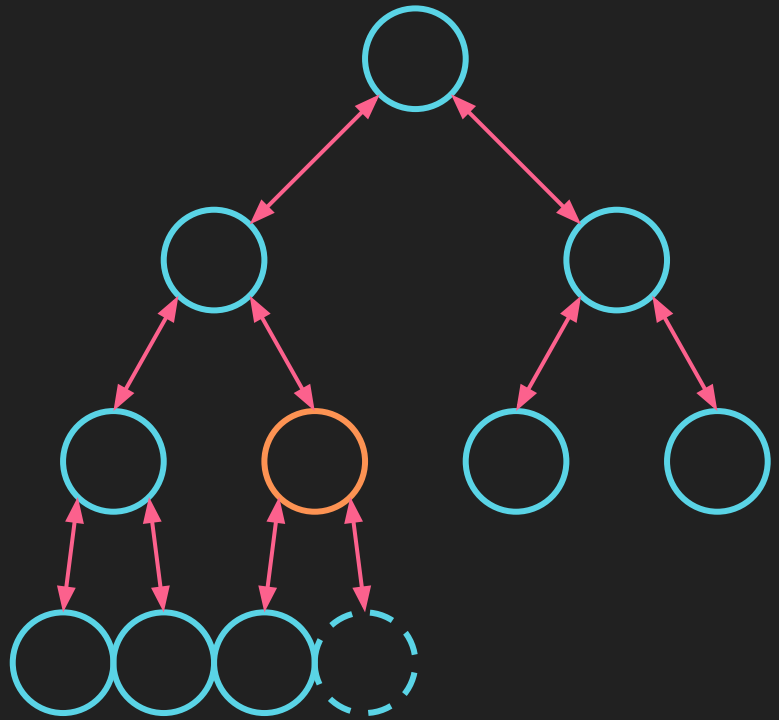
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

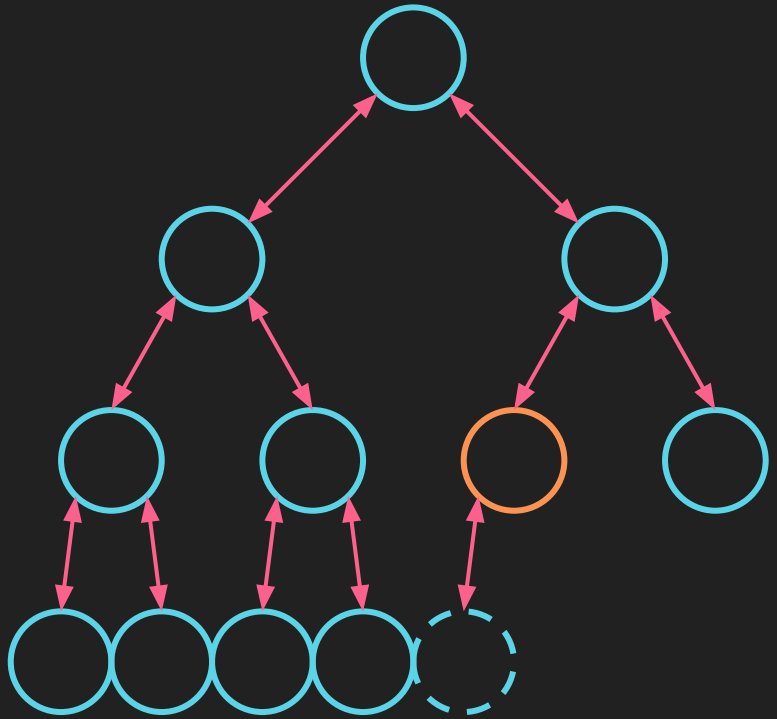
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

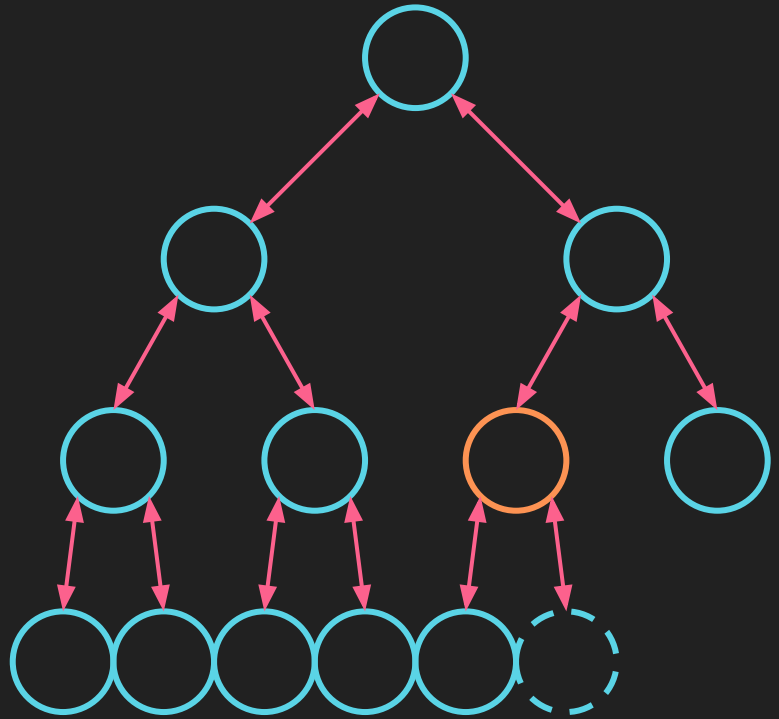
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

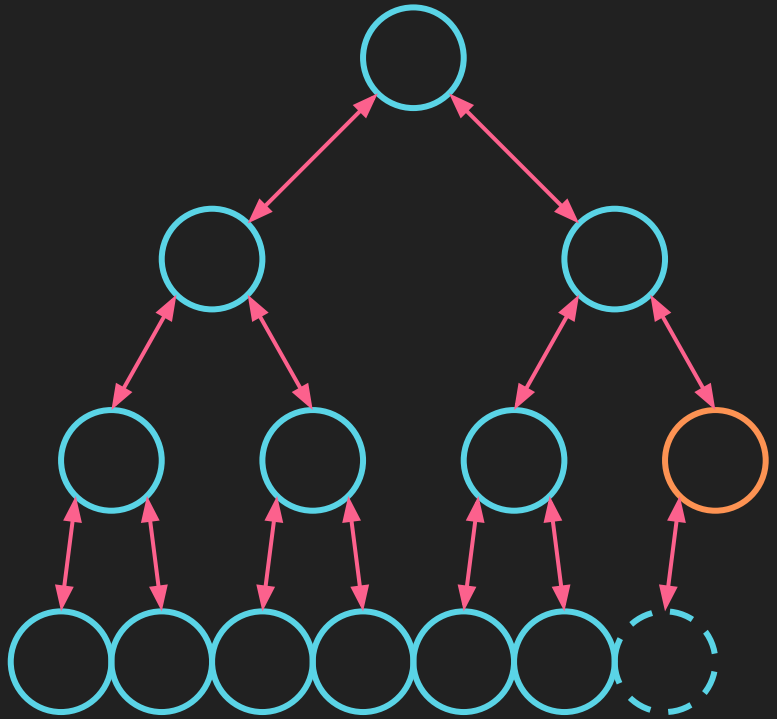
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

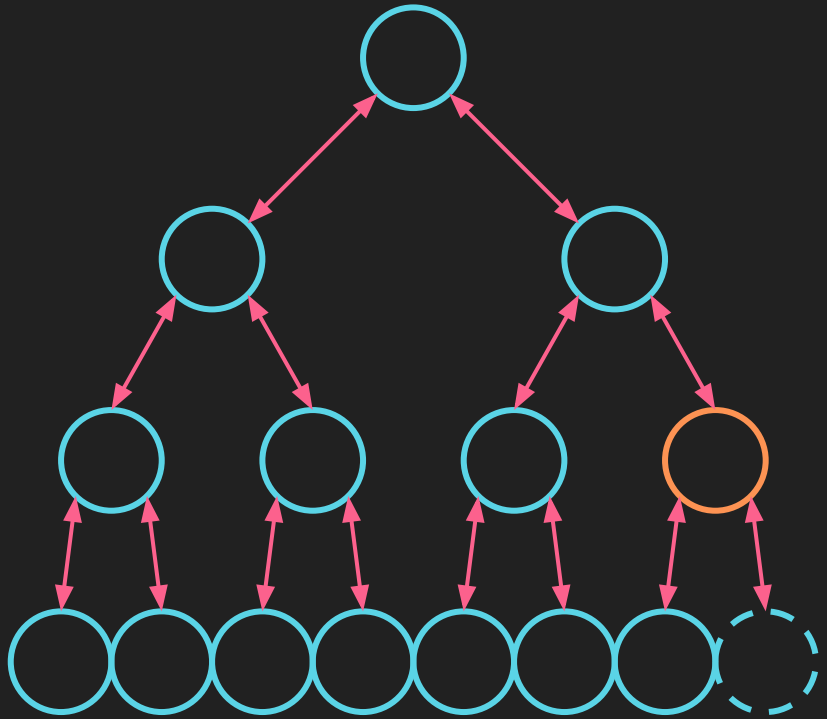
Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

insertion_parent - padre del insertado



Caso 1: Heap no tiene 2 hijos

Caso 2: Ambos hijos están completos y son de la misma altura

Caso 3: El hijo izquierdo está incompleto

Caso 4: Ambos hijos están completos pero son de distinta altura

Caso 5: El hijo derecho está incompleto

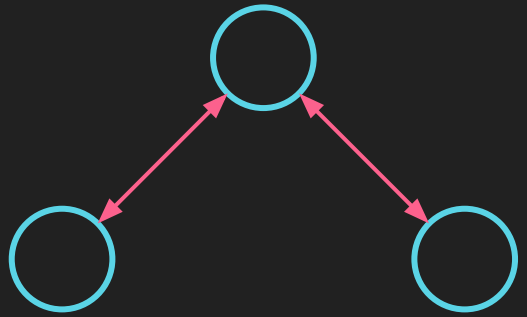
is_complete

```
height = 1

count = 1
```

```
count == pow(2, height) - 1
```

is_complete

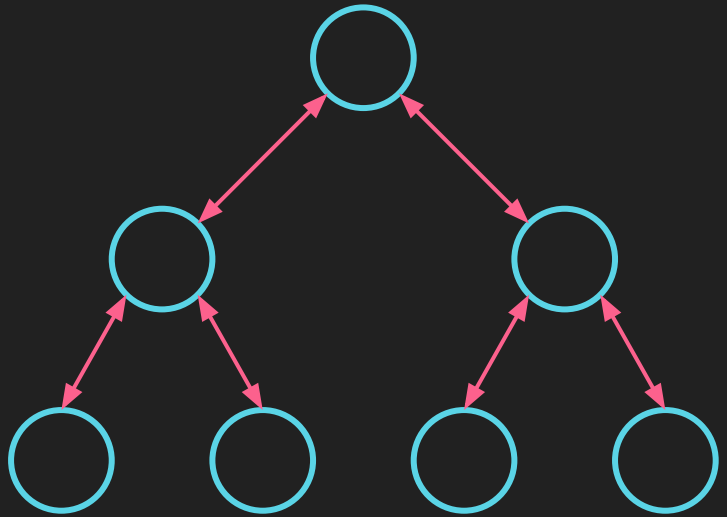


height = 2

count = 3

```
count == pow(2, height) - 1
```

is_complete

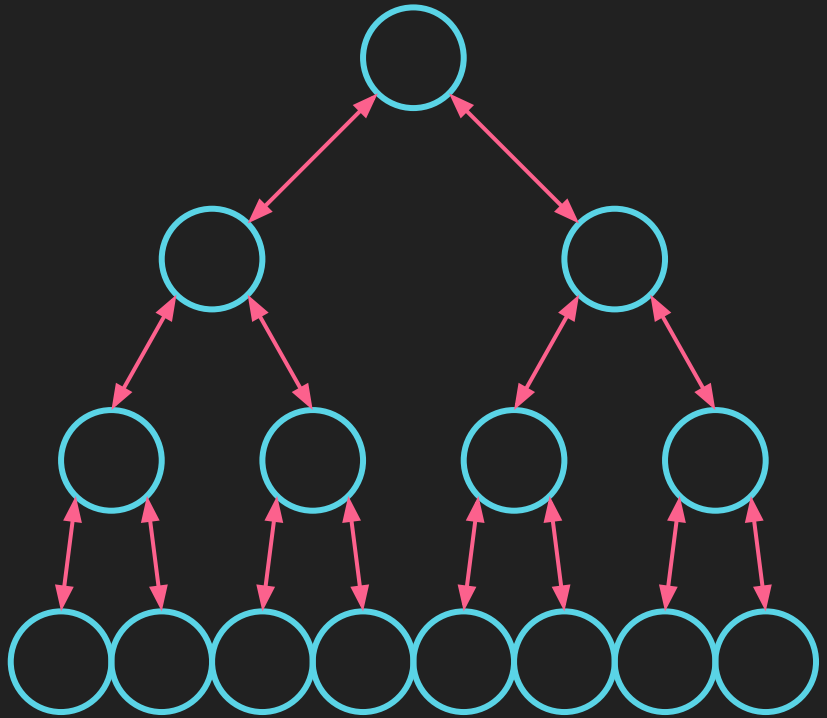


```
height = 3

count = 7
```

```
count == pow(2, height) - 1
```

is_complete



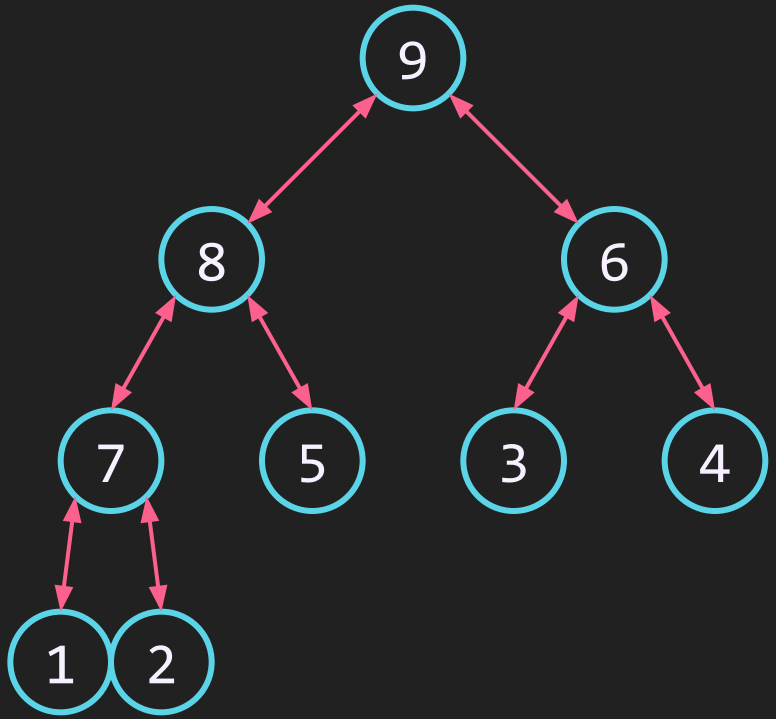
```
height = 4

count = 15
```

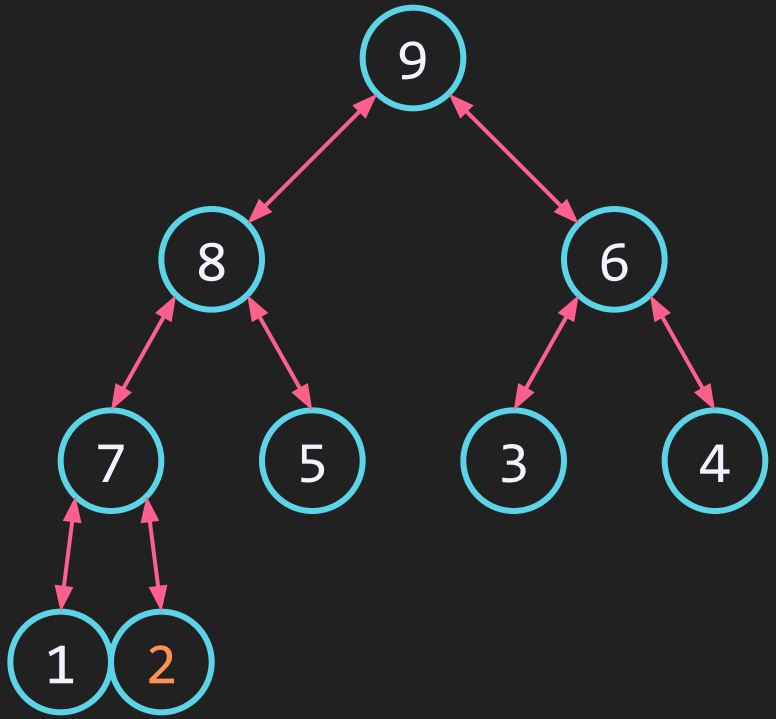
```
count == pow(2, height) - 1
```

heap_extract

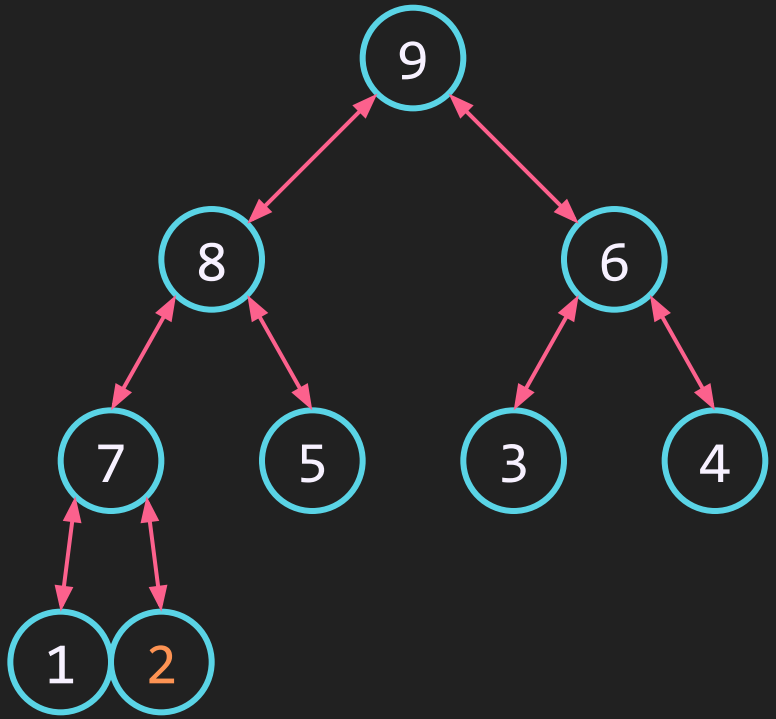
heap_extract(9)



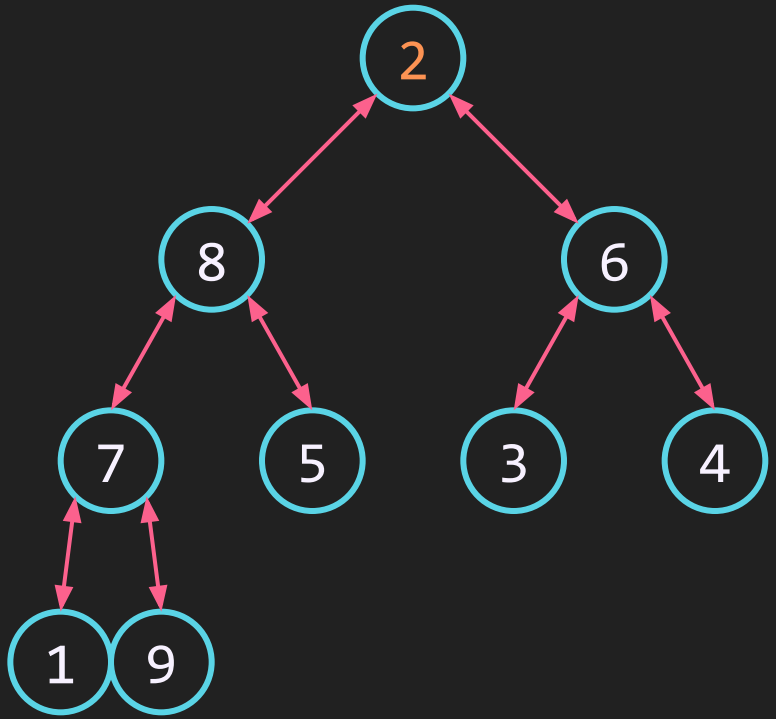
2 = extraction_replacement(9)



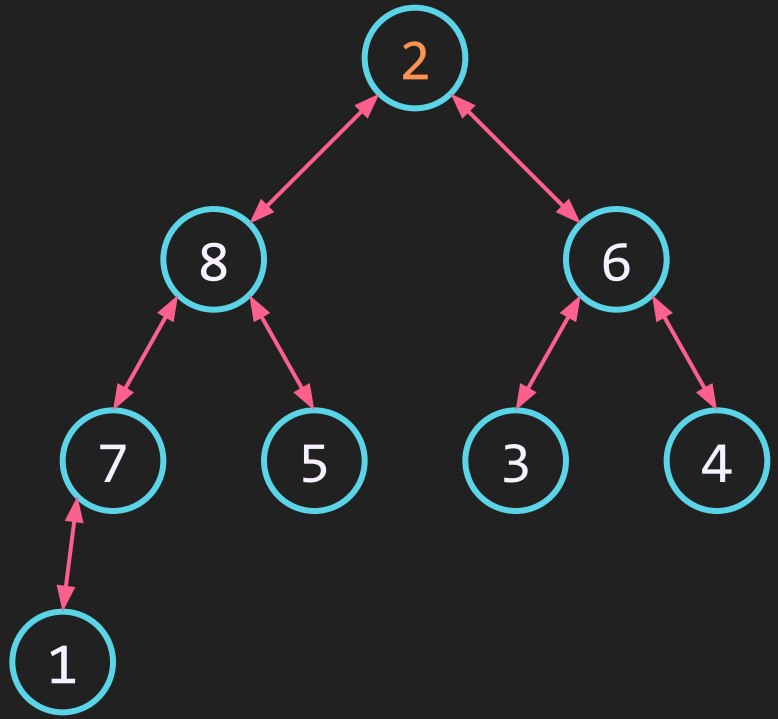
swap(2 , 9)



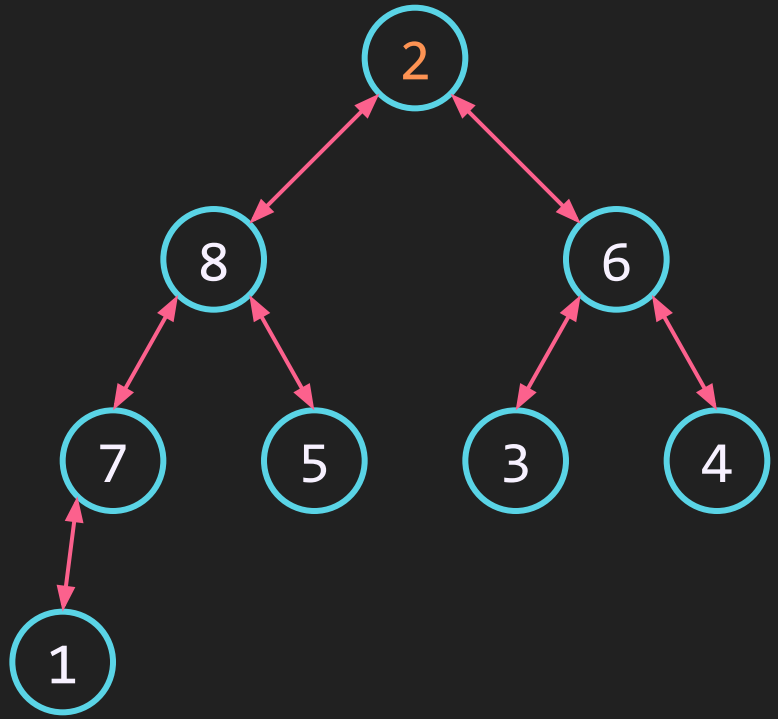
```
best = 9 .priority, delete( 9 )
```



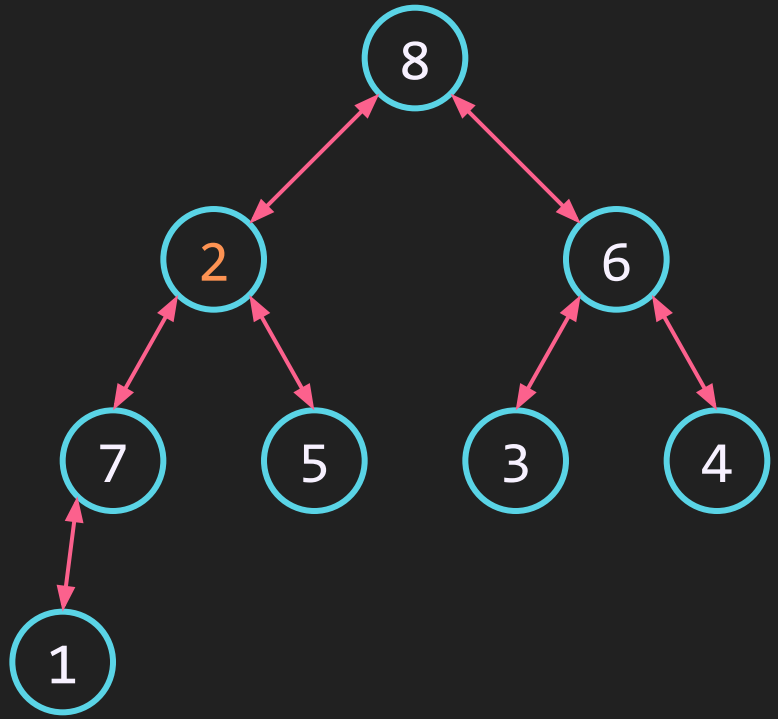
```
best = 9 .priority, delete( 9 )
```



sift_down(2)

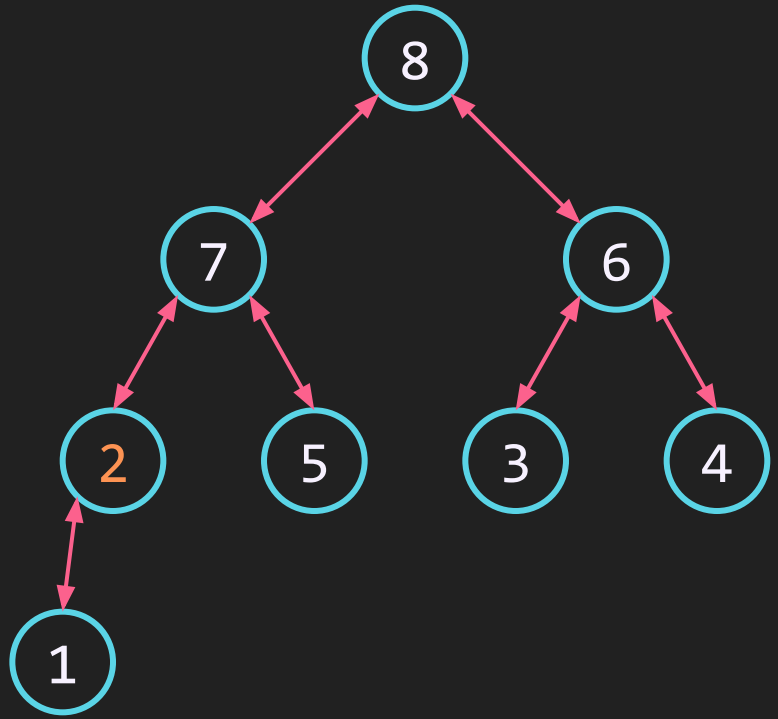


sift_down(2)



sift_down(2)

best



done, return best