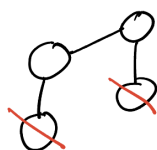


## HW5

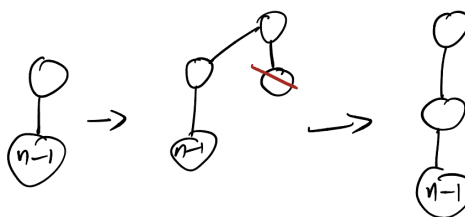
### 林宸昊 PB20000034

- $n = 2$ 。额外增加两个分别比这两个需要插入的数小的数，形成两棵  $B_1$ ，然后调用 `UNION` 操作将两棵树连接；连接之后分别删去两个新增的较小数，得到  $n = 2$  的线性链表；
  - $n > 2$ 。假定现在得到了一棵有  $n - 1$  个结点的线性树，对于需要新插入的数（假定插入前对给定数据已进行排序），再插入一个和这个需要新插入数一样都比当前树结点更小的数，然后 `CONSOLIDATE` 将三者连接起来，然后删去额外引入的数，这样就得到了一棵有  $n$  个结点的线性树。

$n = 2$



$n > 2$



2. (a)

- 这一假设错误。如果  $x$  有不只一个孩子，那么则将子结点加入到  $H$  的根列表所需要的时间则可能不止  $O(1)$ 。教授的假设建立在  $x$  只有一个孩子结点的基础上；
- 实际消耗时间取决于  `$x.degree$` ，事实上对每一个孩子的更新都只消耗常数项时间，因此如果考虑 `CASCADING-CUT` 的调用次数  $c$ ，紧凑上界其实就是  $O(c + x.degree)$ 。