

10:08

- 直接利用展示 node 的 viewport 显示 suggestion 如何

10:37

- 横向打印 node 时混叠问题解决思路:
 - 用一个变量 `curLineWidth` 存储当前行的总长度, 以此与 `viewport` 的宽度比较, 分两种情况:
 1. `m.subselected.y == 0` 时, 若 `curLineWidth > m.viewport.Width`, 则均匀减小每个 node 的字符显示, 如都减小 `n` 个字符, 使得总长不超过 `m.viewport.Width`
 2. `m.subselected.y > 0` 时, 若 `curLineWidth > m.viewport.Width`, 则当前选中的 node 不变, 其他 node 均匀减小字符显示
- 纵向打印 node 时超出问题解决思路:
 - 用一个变量存储当前累加的总高度 `culmuHeight`, `y` 每加一增加一个 `box` 的高度, `y` 每减少一也同时减小一个 `box` 的高度
 - `culmuHeight > m.viewport.Height` 时, 则向下翻页 `m.viewport.ViewDown()`, 然后设置 `culmuHeight = oneBoxHeight`, 注意可能得重新渲染当前 node
 - `culmuHeight == 0` 时, 则向上翻页 `m.viewport.ViewUp()`, 添加一个 `flag` 来判断是不是第一页, 或者计算出 `total page`, 显示在右下角
 - 得到 `culmuHeight` 的计算公式, 在 `UP` 和 `DOWN` 时更新和判断, 增加 `isPageUp()` 和 `isPageDown` 函数