

counter 设置为 0。其中变量 t 代表下落时间，counter 记录循环结构的迭代次数。

脚本随后进入无限循环 ②，每 0.05 秒计算并更新一次小球的位置 ③（让小球流畅地下落），更新下落时间 t 和下落距离 d ，增加变量 counter 的值。

如果小球到达地面（即 $d \geq 35$ ），脚本直接设置其 y 坐标为地面的 y 坐标，然后显示之前计算的下落时间，模拟程序结束 ④。

若小球未到达地面，脚本则根据小球已下落的真实距离（变量 d 的值）设置其在舞台上的竖直坐标位置 ⑤。在本模拟实验的界面中，35 米的高楼对应舞台中 268 个步长（见图 7-29），因此，小球在舞台的下落步长等于 $268 \times (d/35)$ ，最终的 y 坐标即为 136（小球的初始高度）减去该球落下的步长。

由于每次迭代的间隔均为 0.05 秒，迭代 10 次就是 0.5 秒。因此，当循环计数器 counter 等于 10、20、30 等 10 的倍数时，角色 Ball 切换到造型 marker 印一个图章，标记其在下落过程中的位置 ⑥。

图 7-31 展示了本模拟实验的最终效果。你是否发现相同的时间下落距离却不相同？这是因为小球受到地球重力（重力加速度为 9.8m/s^2 ）的作用加速下落。

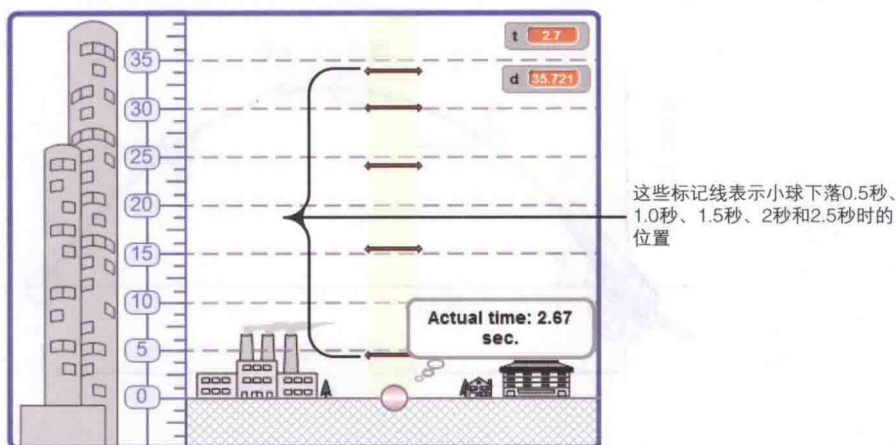


图 7-31：自由落体实验的结果