



图 7-34 : 角色 Ball 的脚本

脚本进入无限循环后 ②，每 0.02 秒计算并更新一次小球的位置。首先计算垂直距离（变量 dy）③。如果该值为负数，说明小球已经到达地面，脚本执行停止当前脚本积木结束程序。

如果 dy 非负，则计算水平距离（变量 d）④。脚本需要把小球实际飞行的水平和垂直距离（d 和 dy）转换成舞台上的 x、y 坐标位置（变量 xPos 和 yPos）。舞台中模拟器的垂直步长为 320（从 -140 到 180），它对应实际中模拟器的 100 米；水平方向步长为 420（从 -180 到 240）也对应实际中模拟器的 100 米。因此，小球在舞台上飞行的垂直距离等于 $320 \times dy / 100$ 米，水平距离等于 $420 \times d / 100$ 步，然后将这两个数值分别加上小球的起始坐标并更新小球的位置。最后，时间变量 t 增加一个较小的数值（本例为 0.02s），并继续迭代小球的下一个位置。

如图 7-33 所示，当小球以 30m/s 的速度斜向上 70° 射出，其飞行时间为 5.75s，射程为 59m。这些数值说明程序非常精准，你甚至可以修改迭代间隔 0.02s 为 0.01s，以提升模拟器的精度，当然轨迹的模拟速度会变慢。因此，你需要调整程序的参数来平衡程序的运行速度和运算精度。