# 系统控制

我们的这一任务集中在 get\_game\_info() 函数中, 它的任务就是简单地调用系统调用, 调整游戏窗口的分辨率以及设置一些系统参数, 以及根据系统调用返回的结果构造相应的 GameInfo 对象.

我们根据不同系统重载了这一函数, 使得在不同的系统上能够使用相同的接口正确地设置游戏窗口和系统, 并得到正确的 GameInfo 对象.

# CI & CD

在github的框架中, CI/CD是声明式的yaml文件, 描述了如何从初始环境开始执行构建, 集成和部署工作. 因此我们将我们的状态图翻译为了对应的文件.

我们利用了github的build matrix功能来在多个不同的平台上并行构建和检查, 以保证我们的工作始终是跨平台的.