详细设计-附表

HIPO

文档编号：详设-？-游戏进行系统

定稿日期：2018/5/18

子系统：游戏进行系统

子系统概述：载入下一场景地图，判断用户是否过关，并进行分数的累加。

负责人：陈帆

实现本子系统需要调动的资源：前端存储的物品库文件，用户的操作。

子系统设计原则：（应参照系统设计原则稍作变动）

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 说明 | 作者 |
| 2018.05.18 | 0.1 | 初稿 | 陈帆 |
|  |  |  |  |

**注：H+IPO图 ——**

***开始工作前请先阅读概要设计文档-附录I-1 体系结构设计报告，尤其是系统总体架构。***

*H代表Hierarchy 层次*

*IPO代表 Input Process Output 输入输出处理*

*所以首先要有这个模块的层次图，再对这个层次图中的各个最小单元进行IPO分析，IPO表中的各项内容解释参考放于同文件夹下的详细设计文稿。*

*简而言之，就是一张结构层次图加上若干张IPO表。*

*下面将附上一份根据这份模板改造的一张IPO图，若不会使用Visio（当然最好是学会，还是挺简单的），可以画一张手绘的层次图然后发给我，由我来代画。*

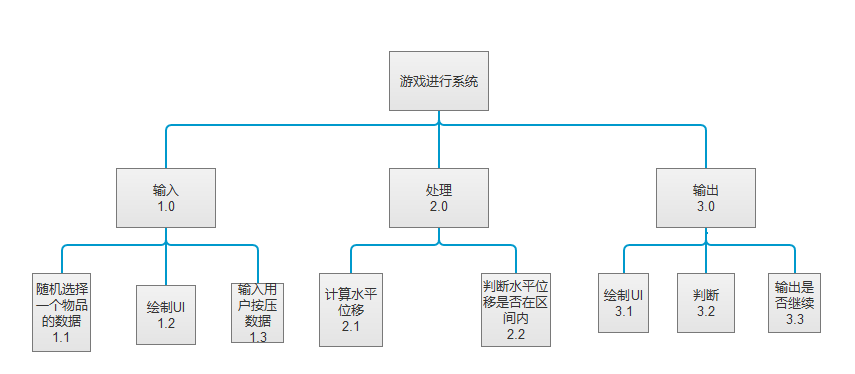
*HIPO根据我的理解主要还是面向过程的一种设计方法，如果不适应可以自行研究面向对象的设计方法提交相应的文档即可。*

*详细设计的方法，详见书本。*

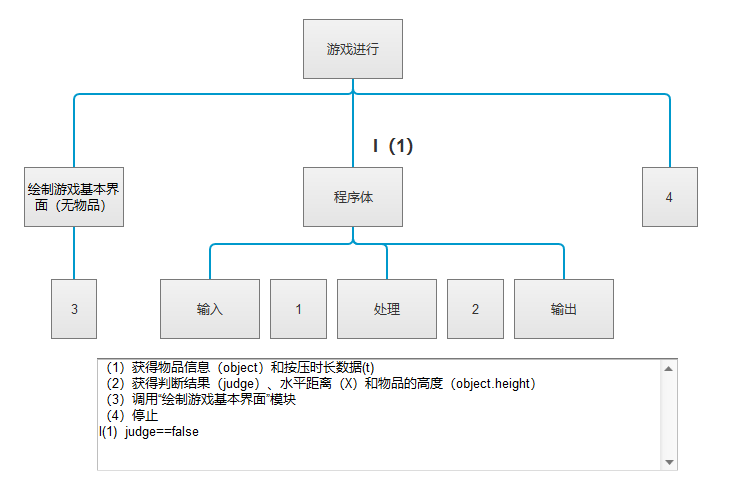
*？个子系统：*

*1、游戏进行系统 -陈帆*

# H图



# 游戏进行系统



伪代码：

游戏进行 seq

绘制游戏基本界面（无物品） seq

调用“绘制游戏基本界面”模块

绘制游戏基本界面（无物品） end

程序体 iter until judge==false

输入 seq

调用输入模块

输入 end

获得物品信息（object）和按压时长数据(t) seq

处理 seq

调用处理模块

处理 end

输入 seq

获得判断结果（judge）、水平距离（X）和物品的高度（object.height）

输入 end

输出 seq

调用输出模块

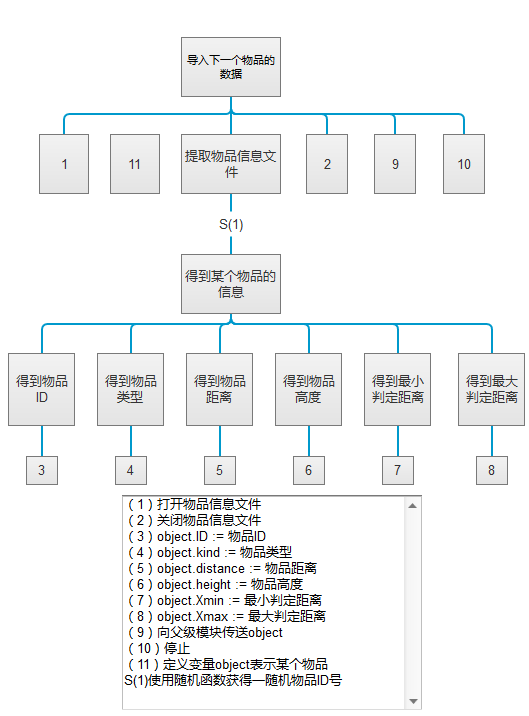
输出 end

程序体 end

停止

游戏进行 end

# 1.1-随机选择一组数据



伪代码：

随机选择一个物品的数据seq

打开物品信息文件

定义变量object表示某个物品

提取物品信息文件 select 使用随机函数获得一随机物品ID号

得到某个物品的信息 seq

得到物品ID seq

object.ID := 物品ID

得到物品ID end

得到物品类型 seq

object.kind := 物品类型

得到物品类型 end

得到物品距离 seq

object.distance := 物品距离

得到物品距离 end

得到物品高度 seq

object.height := 物品高度

得到物品高度 end

得到最小判定距离 seq

object.Xmin := 最小判定距离

得到最小判定距离 end

得到最大判定距离 seq

object.Xmax := 最大判定距离

得到最大判定距离 end

得到某个物品的信息 end

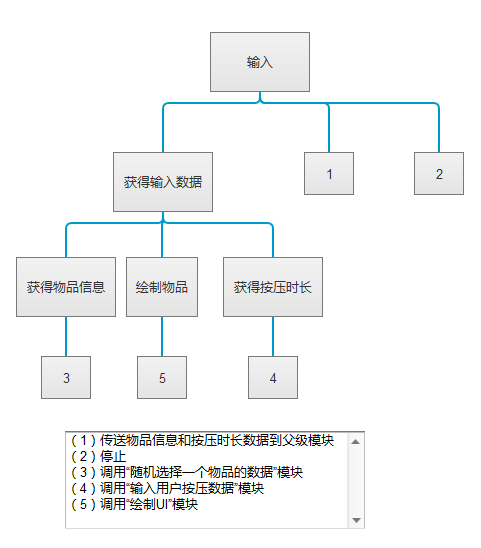
关闭物品信息文件

向父级模块传送object

停止

随机选择一个物品的数据 end

# 1.0-输入



伪代码：

输入 seq

获得输入数据 seq

获得物品信息 seq

调用“随机选择一个物品的数据”模块

获得物品信息 end

绘制物品 seq

调用“输入用户按压数据”模块

绘制物品 end

获得按压时长 seq

调用“绘制UI”模块

获得按压时长 end

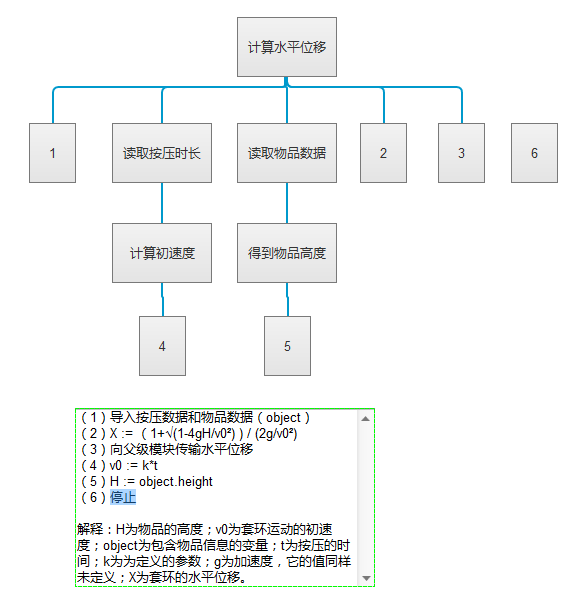
获得输入数据 end

传送物品信息和按压时长数据到父级模块

停止

输入 end

# 2.1-计算水平位移



伪代码：

计算水平位移 seq

导入按压数据和物品数据（object）

读取按压时长 seq

计算初速度 seq

v0 := k\*t

计算初速度 end

读取按压时长 end

读取物品数据 seq

读取按压时长 seq

H := object.height

读取按压时长 end

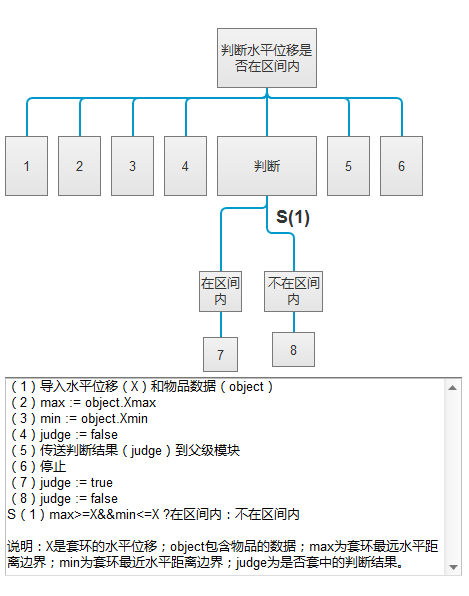
X := （1+√(1-4gH/v0²) ) / (2g/v0²)

向父级模块传输水平位移

停止

读取物品数据 end

# 2.2-判断水平位移是否在区间内



伪代码：

判断水平位移是否在区间内 seq

导入水平位移（X）和物品数据（object）

max := object.Xmax

min := object.Xmin

judge := false

判断 select max>=X&&min<=X

处理在区间内 seq

judge := true

处理在区间内 end

判断 or max<X||min>X

处理不在区间内 seq

judge := false

处理不在区间内 end

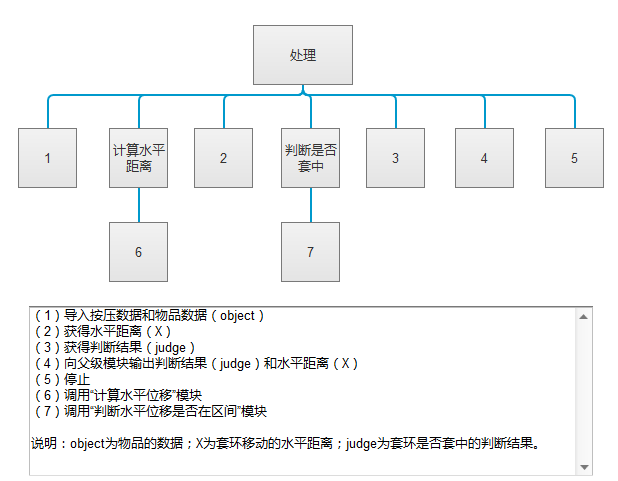
判断 end

传送判断结果（judge）到父级模块

停止

判断水平位移是否在区间内 end

# 2.0-处理



伪代码：

处理 seq

导入按压数据和物品数据（object）

计算水平距离 seq

调用“计算水平位移”模块

计算水平距离 end

获得水平距离（X）

判断是否套中 seq

调用“判断水平位移是否在区间”模块

判断是否套中 end

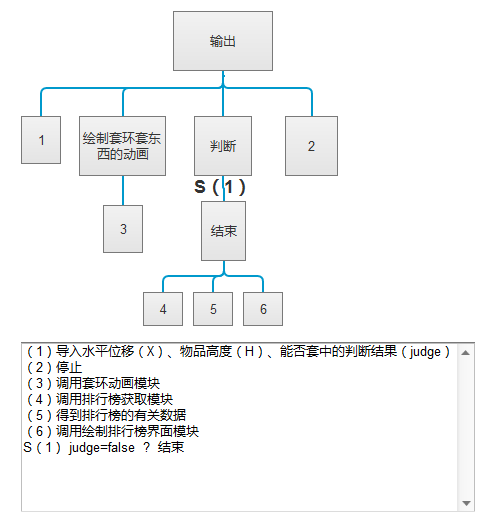
获得判断结果（judge）

向父级模块输出判断结果（judge）和水平距离（X）

停止

处理 end

# 3.0-输出



伪代码：

输出 seq

导入水平位移（X）、物品高度（H）、能否套中的判断结果（judge）

绘制套环套东西的动画 seq

调用套环动画模块

绘制套环套东西的动画 end

判断 select judge=false ? 结束

结束 seq

调用排行榜获取模块

得到排行榜的有关数据

调用绘制排行榜界面模块

结束 end

判断 end

停止

输出 end