Insaniquarium 游戏说明文档

软件 72 林芳云

一、游戏简介

Insaniquarium（中文名：怪怪水族馆）是流行桌面游戏 Insaniquarium！Deluxe（中文名：怪怪水族馆！豪华版）的开源 Qt 复刻版本。相比原版本，Insaniquairum 实现了基本的功能与逻辑，保留了原游戏的核心玩法，并提供了可扩展的框架。

二、安装运行

游戏为绿色安装版本。解压缩后点击 bin 目录下的.exe 文件即可运行。
*需要 Windows7 以上的 64 位系统和 Microsoft VC++2015 支持。

三、游戏背景

若你以为当个宠物主人是一件容易的事情，那你可就错罗！不但要确保自己的心爱宝贝是否有吃饱，还得要注意是否有外物入侵，Insaniquarium 怪怪水族馆，这回儿要让你养鱼养到傻眼，连外星人都来参一脚，美人鱼也来凑热闹！这到底是怎么回事呢？怪怪水族馆，是一个超级好玩的养鱼游戏，此水族箱养着各式各样的鱼儿，美丽的画面让人赏心悦目，但是，要让那些鱼儿乖乖的吃饱饱，饲主可得要花心思了，不但鱼饲料要花钱买，各式各样的状况还有不同的解决方式，万一有外星人在水族箱里头打算吓走鱼群，饲主还得要出面将外星人打跑，怪怪水族馆，是一个挺有趣的游戏，异于一般的养宠物，此游戏有更多的花招，帮你打发时间！

四、 游戏角色介绍

鱼

1、 古比鱼：分成小（售价$100）、中、大、王四种体型，以饲料为食。小、中古比鱼吃到足够多的饲料后可以立刻变成中、大古比鱼。大古比鱼在长时间后有概率变成古比鱼王。小古比鱼会产出银币，中、大古比鱼会产生金币，古比鱼王产出钻石。
2、 古比妈咪：分成中、大 2 种体型，以饲料为食。中古比妈咪吃到足够多的饲料后可以立刻变成大古比妈咪。可以定期生出一只小古比鱼，但是大古比妈咪生产的速度是中古比妈咪的 2 倍。
3、 食肉鱼：售价$1000，以小古比鱼为食，产出钻石。

4、巨型食肉鱼：售价$10000，以食肉鱼为食，产出宝箱。

外星人

1、蓝色水怪(角色名：Sylvester)：通过撞击鱼类使其致死。鼠标点击会使其掉血，并向点击处的反方向躲避。
2、贪吃怪(角色名：Gus)：会吃掉鱼类。吃饲料和鱼类会使其掉血，鼠标攻击对其无效，运动方向不受玩家控制。
3、红毛怪(角色名：Balrog)：通过撞击鱼类使其致死。鼠标点击会使其掉血，并向点击处的反方向躲避。

宠物

1、蜗牛(角色名：STINKY)：第二关出场，在地上左右爬行，捡玩家掉落的金币。
2、旗鱼(角色名：ITCHY)：第三关出场，会尾随并攻击外星人。
3、海马(角色名：ZORF)：第四关出场，会吐出鱼食喂养鱼。
4、鱼骨头(角色名：VERT)：第五关出场，会产出金币。
5、河蚌(角色名：CLAM)：第六关出场，会定期打开，内含一颗珍珠，可以被玩家点击获取。

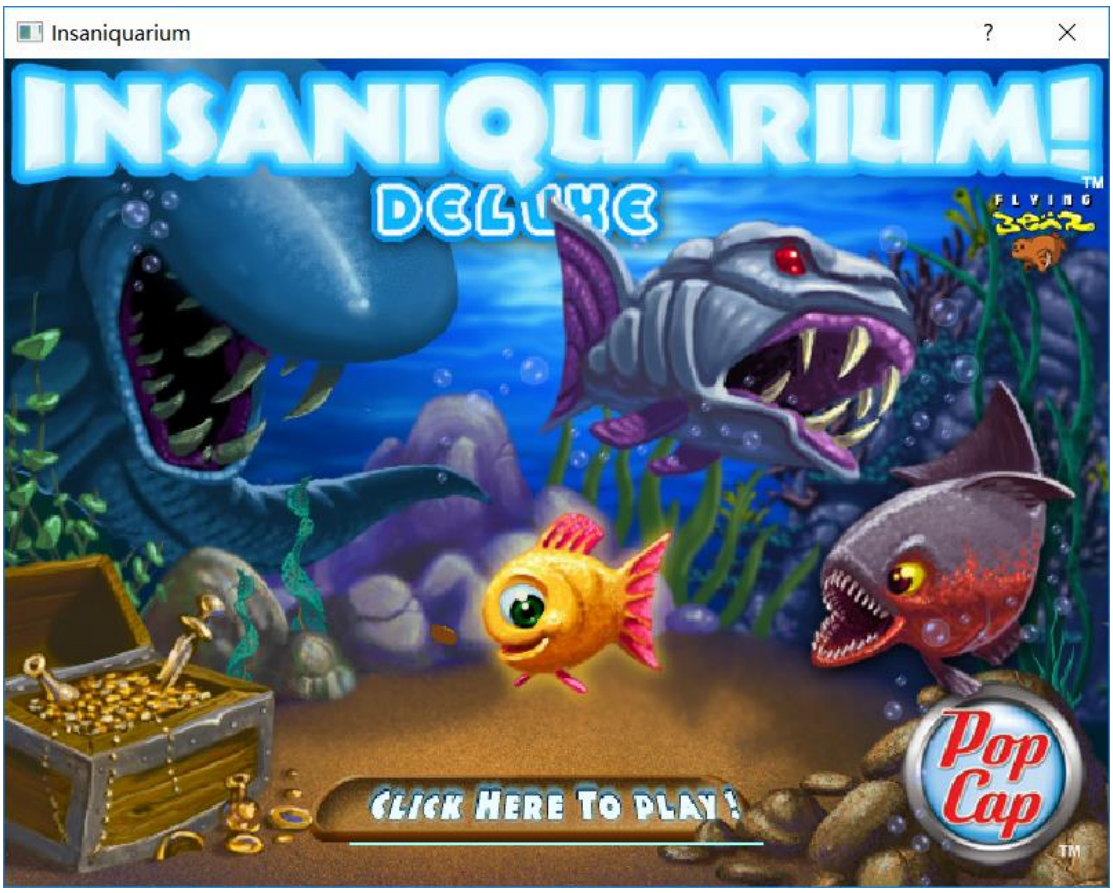
宝物

1、银币：由小古比鱼产出，价值$15。
2、金币：由中、大古比鱼，鱼骨头产出，价值$35。
3、钻石：由食肉鱼、怪物死亡、古比鱼王产出，价值$200。
4、珍珠：由河蚌产出，价值$200。
5、宝箱：由巨型食肉鱼产出，价值$10000。

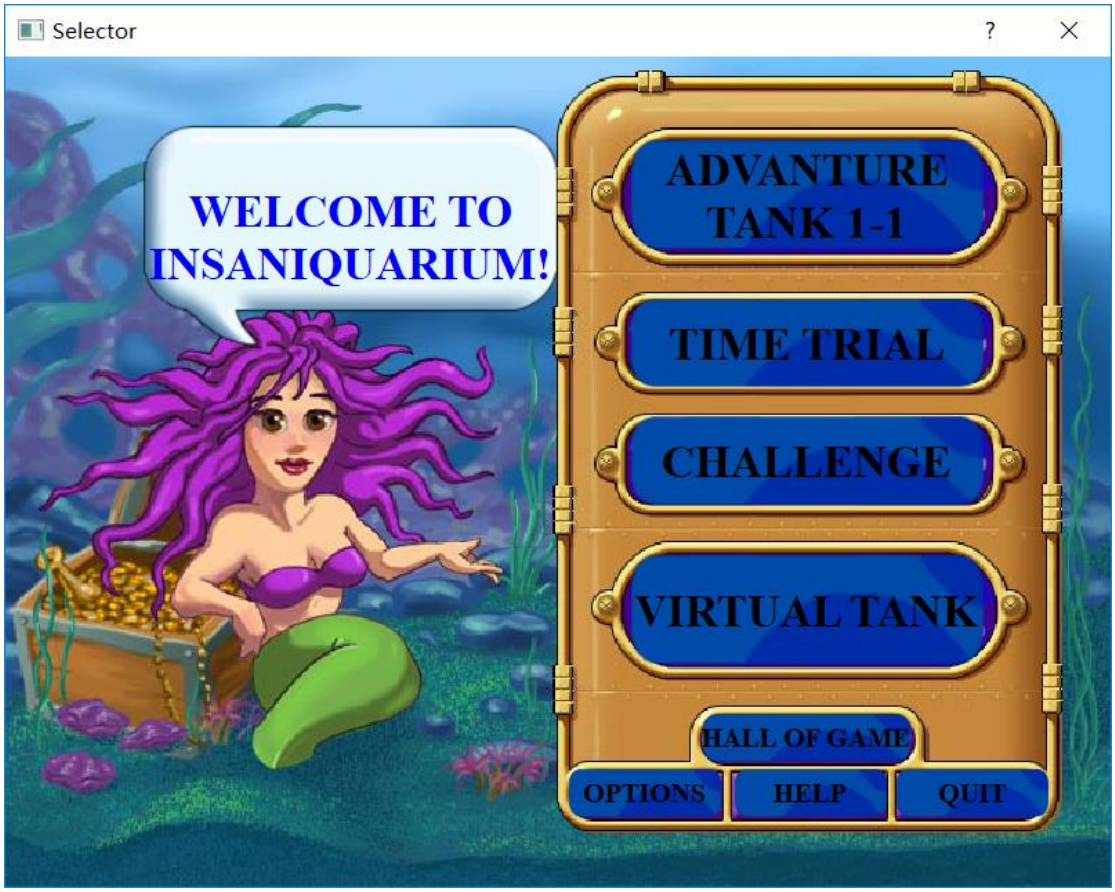
食物

1、初级饲料：售价$5。
2、二级饲料：通过升级初级饲料获得，售价$5，升级费用$200。
3、三级饲料：通过升级二级饲料获得，售价$5，升级费用$300。

五、 游戏界面


Insaniquarium
INSANIQUARIUM!
DELUXE
FLYING BEAR
CLICK HERE TO PLAY!
PopCap

主菜单


Selector
WELCOME TO INSANIQUARIUM!
ADVANTURE TANK 1-1
TIME TRIAL
CHALLENGE
VIRTUAL TANK
HALL OF GAME
OPTIONS
HELP
QUIT

单击 HELP 弹出游戏规则

进入游戏

右上角显示分数。

分数超过 100：上方第一个窗口打开，可以购买小古比鱼。

分数超过 200：上方第二个窗口打开，可以将鱼食从 1 级升级到 2 级。

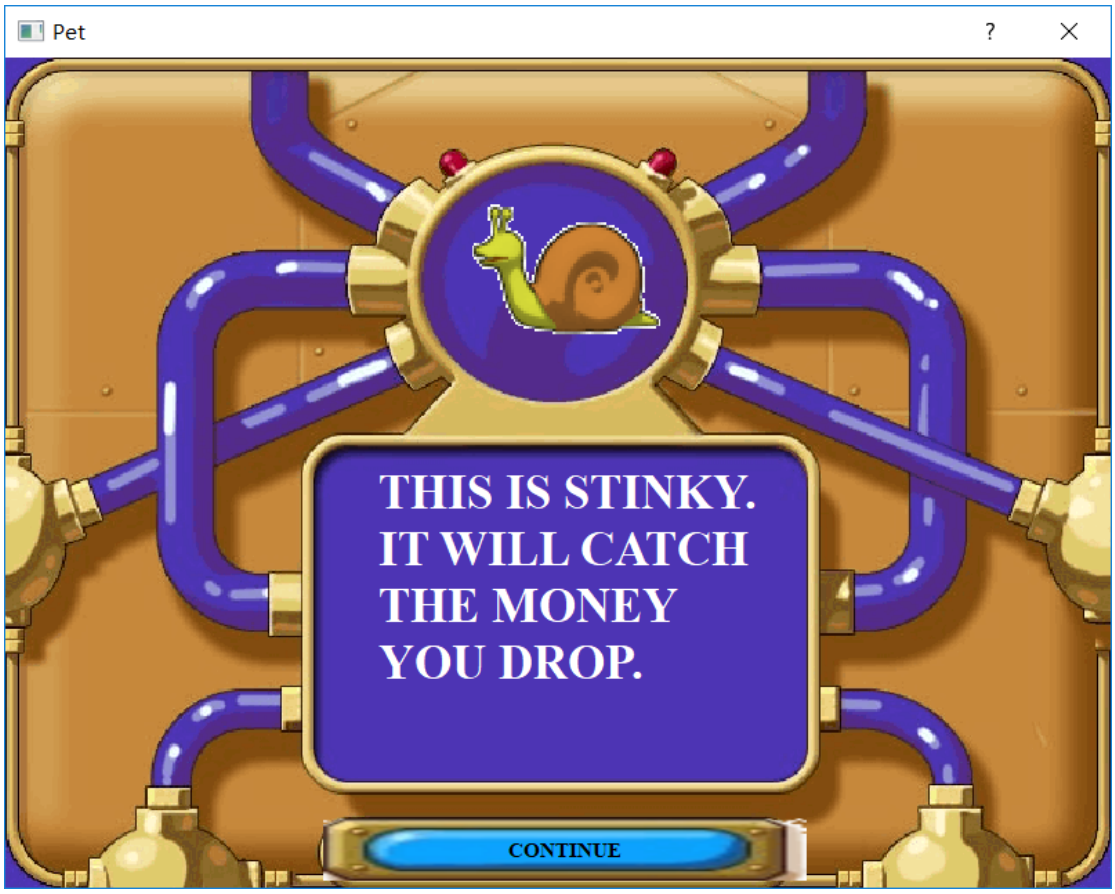
分数超过 300：上方第三个窗口打开，可以增加鱼食数量。第二个窗口打开，可以将鱼食从 2 级升级到 3 级。

分数超过 1000：上方第 4 个窗口打开，可以购买食肉鱼。第 7 个窗口打开，可以收集一片蛋壳碎片。

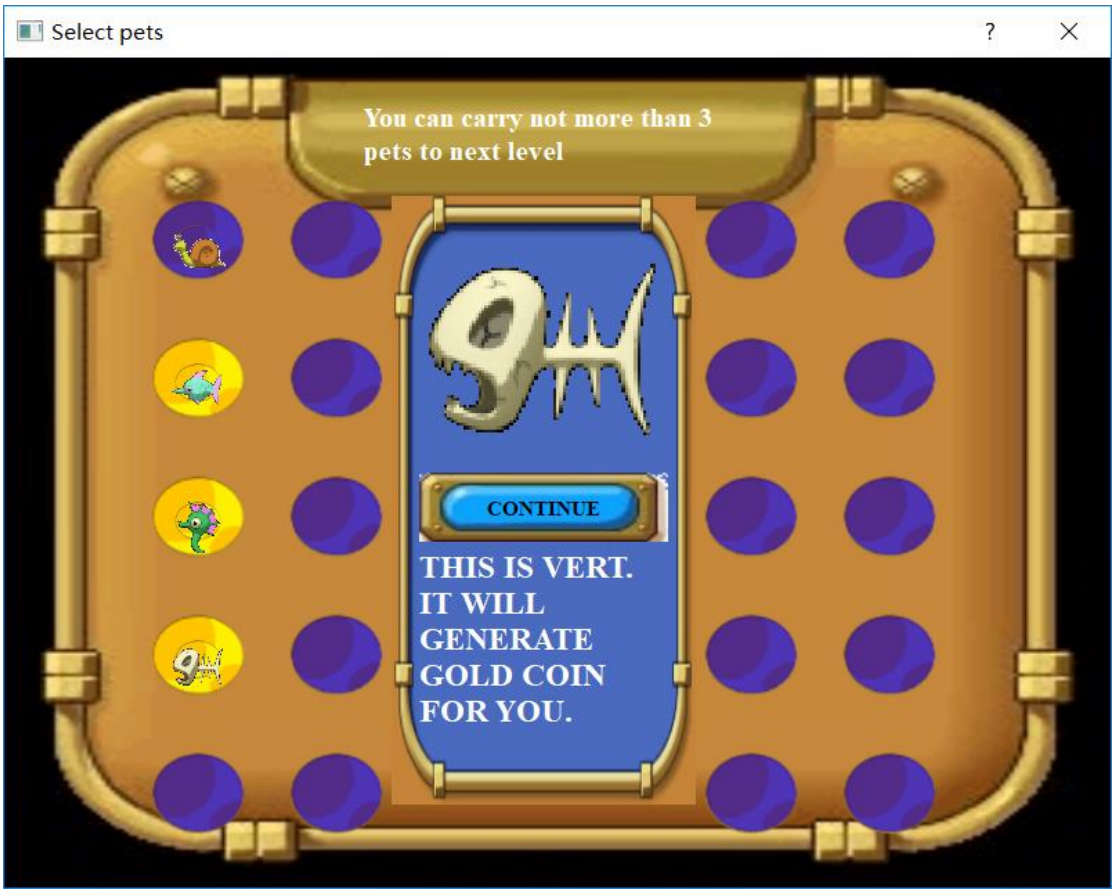
分数超过 10000：上方第 5 个窗口打开，可以购买巨型食肉鱼。

窗口部分打开后如图所示：

收集满 3 个蛋壳碎片后，游戏自动通关，并介绍下一关的宠物。



每一关结束后获得一只宠物。如果获得宠物数量>3，会出现宠物列表界面，要求挑选最多 3 只宠物进入下一关。



六、 其他功能

1、 界面语言切换

在主菜单点击 OPTIONS，弹出的设置界面中可以选择使用中文/英文。

2、 游戏进度的保存

退出游戏，重新进入游戏，显示“欢迎回来/WELCOME BACK”的字样。游戏的关卡数、分数、所有游戏角色的状态都复原到退出之前的情况。

3、 重置游戏进度

在主菜单点击“OPTIONS/设置”，在弹出的对话框中点击 RESET。将会清空玩家的一切游戏记录，从第一关重新开始。同时，玩家的个人信息也将不再保存，因此需要重新输入用户名。

4、 游戏界面右上方的 MENU 按钮

单击 MENU,弹出对话框。

选择“RESTART/重来”：从当前这一关的最初开始。

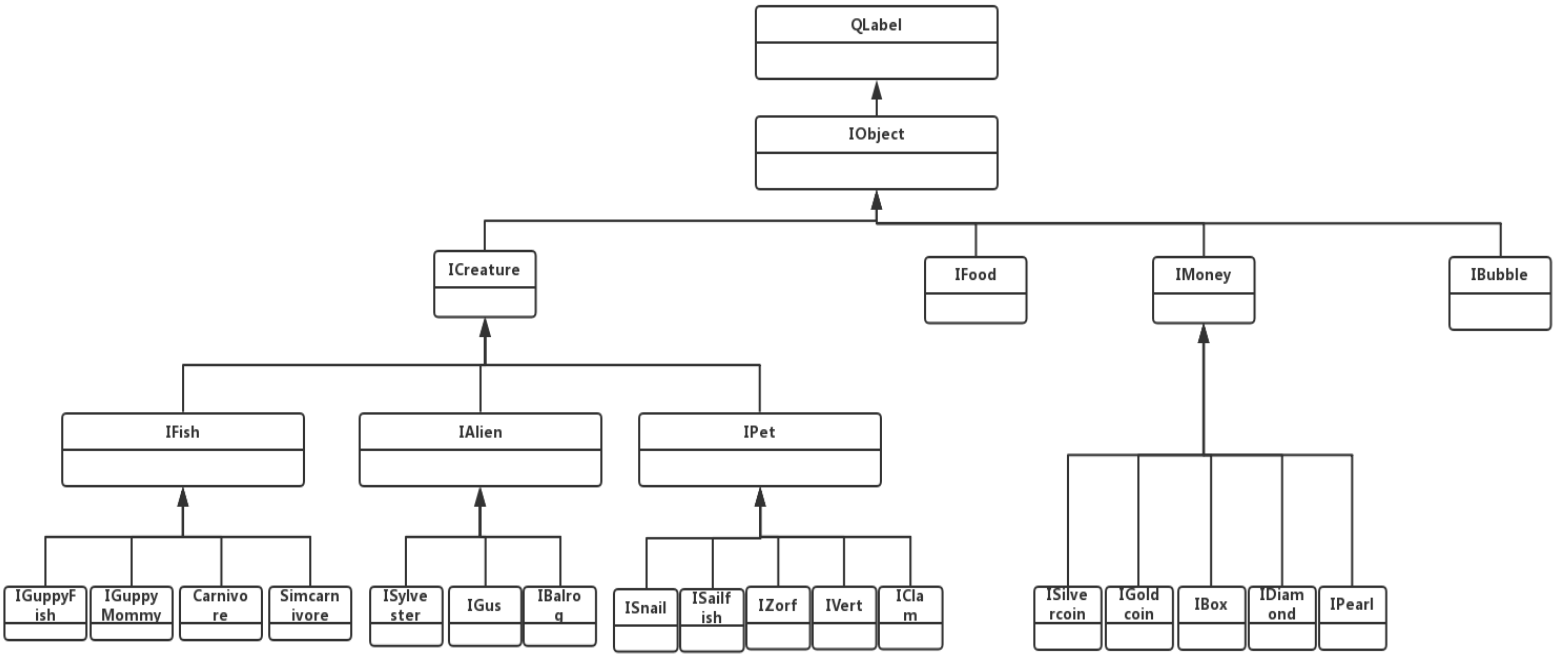
选择“BACK TO MAIN MENU/回到主菜单”：回到主菜单

选择“QUIT”：退出

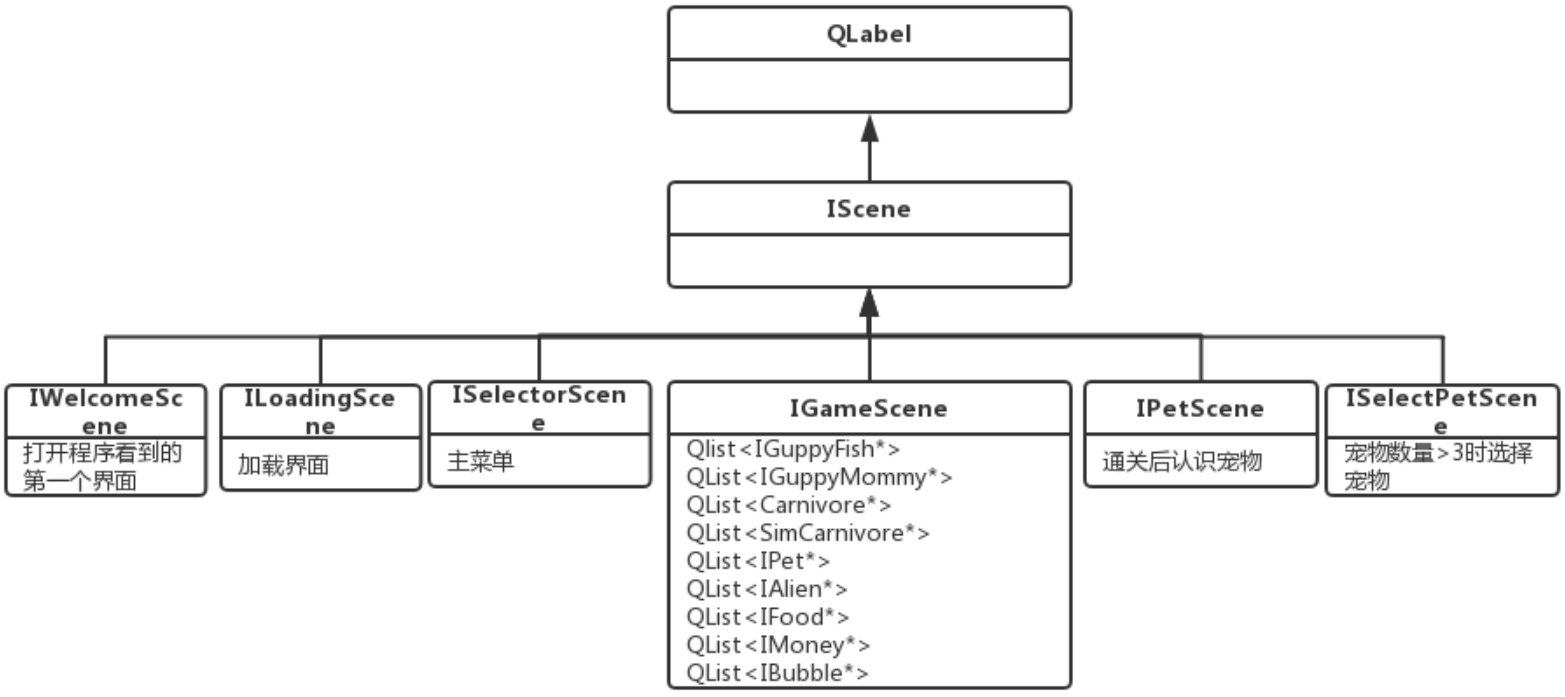
选择“CONTINUE”：游戏继续。

七、 程序架构

物体类



场景类



窗口类

MainDlg 类：继承自 QDialog，在 main.cpp 中实例化通过 show 显示。
成员变量中有一个 IScene*类型的 scene 指针，通过 new 操作实例化一个场景类。如需切换场景，则 scene->deleteLater();scene=new …即可。MainDlg 对象和通过 scene 实例化的场景对象之间是父子关系。

辅助类

IGameData 类：负责记录静态的游戏数据，如每个宠/怪物的编号，每关获得宠物的编号，每张贴图的尺寸（游戏贴图的大小和实际角色大小并不贴合）。
Mybutton 类：自定义按钮，继承自 QLabel，可以设置 hover,pressed,leave, normal 四种状态的图片，来模拟按钮被按下的效果。
MybuttonMulti 类：按钮组中的单个按钮，需要记录是否被选中。主要用于选择宠物界面。

运行逻辑

MainDialog 类负责界面的显示，场景的切换，记录一些与游戏内容无关的数据，如用户名、界面语言等，接受来自子对象即各个场景的信号并处理，如更改界面语言等。
继承 Scene 类的各种场景类负责场景的显示，以及各自具体的功能。IWelcomeScene 向其父对象即 MainDlg 递交用户输入的姓名；ILoadingScene 没有特殊功能；ISelectorScene 接受用户提交的切换界面语言的请求、重置游戏数据的请求，并递交给 MainDlg 类；IPetScene 没有特殊功能；ISelectPetScene 向 MainDialog 类递交用户选择的宠物编号；IGameScene 十分复杂。

游戏逻辑

游戏逻辑全部在 IGameScene 中完成。IGameScene 类有一个一个 QTimer*成员变量，每隔 20ms 到期一次，执行如下逻辑：

a) removeDeath：除去所有已经不存在的游戏角色
   注意：不存在≠死亡。当一条鱼饿死后，虽然已经死亡，但是还要漂浮一会儿，所以只有当其沉到地面时才能视为不存在。
   通过模板函数 removeDeathByGroup 依次检查 IGameScene 中的每个 QList，将其中被标记为“不存在”的对象内存释放即可。
b) checkAlien：检查外星人是否降临。
   普通的游戏角色无法感知外星人的到来，只有通过游戏场景检测外星人，并将检

测结果通知所有游戏角色。此处用到了观察者模式的思想。被通知外星人到来后，鱼类的饥饿时间会停止增长（也就是外星人降临期间普通的鱼类不会饥饿，饥饿的鱼类不会饿死），蜗牛会缩进壳中，旗鱼会攻击外星人。

c) shift：执行每个游戏角色的逻辑。

一般而言，每个游戏角色，无论是否为活物，都要执行这样的游戏逻辑：

① 判断是否撞墙，如果撞了，变换速度和

② 做自己特有的事情

③ 如果有必要，更新贴图

④ 更新位置

其中，①④两步是在 IObject 中写好的函数，被所有子类继承。

鱼类的②一般还要包括 checkFood（检查周围是否有食物，如果有，吃掉；如果没有，朝着最近的食物游过去）。checkFood 可以在所有具体鱼类的父类——IFish 类中以模板函数的方式实现，各种鱼类对象只需要传入能吃的食物的列表作为参数即可调用。

更新贴图只有在运动方向、状态（正常、饿、死亡）、大小改变的情况下才需要进行，在调用之前要做判断。一般而言，更新贴图会在 IFish 类、IAlien 类和 IPet 类这些抽象类中而非其子类中实现，子类只要直接调用就好，因为属于同一个父类的游戏对象的贴图都有统一的命名标准，例如所有的鱼类命名都是【前缀+动作+状态+方向.gif】，比如 xxxxxx/eatnormalleft.gif。前缀是父类的成员变量，被子类继承并赋值为各自的路径，在子类调用父类函数时会动态地赋值为子类的前缀。

d) Judge：判断游戏是否终止：通关 or 鱼死光。

进度的读取、保存

每个游戏对象都有一个 read 和 write 函数，参数为 const QJsonObject& json。在 read 中，通过 data=json["data"].toInt()的形式读取数据；在 write 中，通过 json["data"]=data 的形式写入数据到 json。

在 IGameScene 中有一个 write 和 read 函数，参数为 const QJsonObject& json。在 write 中调用每个游戏对象的 write，参数为 json；在 read 中，读入 json 中的数据，按照键名分发到每个游戏对象的 read 函数中。read 函数被封装在 loadGame 函数中，loadGame 读入一个 json 文件，将其中的内容转换为一个 QJsonObject，作为参数调用 read 函数；write 函数被封装在 saveGame 函数中，流程同上。

调用 loadGame 的时间：用户有游戏记录但是第一次打开游戏；用户从游戏界面返回主菜单然后又回到游戏。由于这两种情况都是从主菜单进入游戏界面，所以易于标识。

调用 saveGame 的时间：在游戏状态关闭窗口；从游戏界面返回主菜单。

MainDlg 会向 json 文件中写入界面语言和用户名，并在游戏加载时读入语言和用户名。通过 json 文件中是否存在古比鱼的记录，判断用户是否第一次玩或者重置了数据，如果是，

那么 MainDlg 会向 json 文件中写入一些默认的游戏参数，比如 level=1, score=100, language="en"，供 IGameScene 在 loadGame 中加载。
数据文件名为 save.json，位置为当前目录。如果运行.exe 文件，就在 bin 目录下；如果打开工程文件编译运行，就在 build 目录下。

八、总结

记录一下印象深刻的 bug 们：

1、 delete 和 deletelater 不能混用。比如在父对象中使用 delete，子对象中使用 deletelater，就是一种很危险的行为——delete 是立刻释放内存，但是 deletelater 是当前对象处理完了自己的信号和槽函数之后再将其释放。当父对象被 delete 后顺便 delete 了自己的子对象，而子对象还有信号和槽要处理时，常见的 read access violation 就出现了。

2、 写一个文件有 2 种方式：Append 和 Truncate，然而打开 json 文件不要用第一种。
假如原本 json 文件中是这样的内容：

```
{
    "item1":{ balabala },
    "item2":{ balabala }
}
```

用 Append 方式打开并 json["item3"]=balabala 后的效果：

```
{
    "item1":{ balabala },
    "item2":{ balabala }
}
{
    "item3":{balabala}
}
```

所以应该用 truncate 方式打开，先将文件内容读入一个 QJsonObject json，然后 json.insert("item3", balabala)，就可以达到预期的效果：

```
{
    "item1":{ balabala },
    "item2":{ balabala },
    "item3":{balabala}
}
```

3、资源文件是只读的，所以数据文件不能放进资源，只能放在同级目录了。

4、QList 的 clear 是不释放内存的，需要 qDeleteAll。

5、QSingleShot 不要用在随时可能被终结的对象里，比如鱼、外星人。因为 QsingleShot 要执行的事件不会因为对象被 delete 而中断，从而出现耳熟能详的 read access violation。

6、connect(button, SIGNAL(clicked), xxxxx)和 eventFilter 不能混用，不然前者会失效，因为 button 发出的信号已经被 eventFilter 截获了。

7、lambda 表达式十分方便，但是只能写成 connect(button, &QPushButton::clicked, xxxx)，不能写成 connect(button, SIGNAL(clicked), xxxx)。

8、ui->retranslate(this)可以实现方便快捷的国际化，但是只对直接显示在 ui 上的字体有效。如果 ui 上放了一个自定义的 dialog，那么这句话对 dialog 上的所有控件上的字体都不起效。