

买刀买马 (Knife and horse)

——the veg

一、游戏介绍

由 the veg 开发的游戏“买刀买马”是一款 2-8 人参与的联机策略对战游戏，主要素材来源于“我的世界”。游戏的目标是通过策略和战略，控制角色与其他玩家进行对战。玩家可以购买各种刀具和马匹来增强角色能力，并利用地形来制造优势。游戏的胜利条件是击败其他玩家。

游戏规则如下：

1、开局每人选择一个职业：

①末影人：无买马权，有买珍珠权 1 个(消耗 2 步)，无法使用马，有珍珠后，移动一步可以到任何地方，当有两个珍珠时，每回合结束可以额外移动一次。

②狂战士，无买刀、马权，无衣服，无法使用刀、马、衣服，有买石斧权 1 个(消耗 1 步伤害为 1)，买铁斧权 1 个(消耗 2 步伤害为 2)，买钻斧权 1 个(消耗 3 步伤害为 3)，当有两个同种斧头时，伤害乘二。

③女巫：有买药水权无数个(消耗 1 步回复 1 滴血)，有买喷溅药水权无数个(消耗 1 步延时在某地回复 1 滴血)。

④弓箭手：有买弓权 1 个(消耗 2 步伤害为 1)，有买箭权无数个(消耗 1 步)，即射箭需要买弓买箭射箭，距离无限，当有两个弓时，射箭伤害为 2。

⑤炮手：有买炮 1 个(消耗 2 步伤害为 2)，有买炮弹权无数个(消耗 2 步)，即射炮需要买炮买炮弹发射，距离无限，当有两个炮时，发射伤害为 4。

⑥雷兵：有买雷权无数个(消耗 1 步伤害为 1)，埋雷消耗 1 步，一次性引爆所有你埋的雷消耗 1 步。

⑦胖子：无买马权，不能使用马，有一件无法被抢夺与继承的衣服，移动时可携带位于同一位置的某人。

⑧气功大师：无买刀、马权，无衣服，无法使用刀、马、衣服，有买石拐权 1 个(消耗 1 步伤害为 1 只能攻击同一位置的人)，有买铁拐权 1 个(消耗 2 步，伤害为 1 能攻击任意位置的人)，有买钻拐权 1 个(消耗 3 伤害为 2 只能攻击在同一位置的人，使用时拐杖强制将使目标移动一步)，当有两个同种拐杖时，伤害乘二。

⑨吸血鬼：每使用刀攻击一次，回复 1 滴血。

2、每人血量为 10，装备初始为 1 件衣服，买刀权 1 个(消耗 1 步伤害为 1)，买马权 1 个(消耗 1 步伤害为 3，只能在两人位于同一座城时使用，使用后目标将被踢到中心区域)，开局位于自己的城内，所有人城外为同一个中心区域。

3、每回合通过剪刀石头布每个人赢的人数决定每人回合开始行动步数。

4、每人的每一步行动必须选择下列行为中的一种：

①移动:在城中则走到中心区域，在中心区域则走到某个城内。

②购买:拥有买刀权或买马权，可选择买刀或买马并消耗买刀权或买马权。

③抢夺:当你与某个人位于同一位置时，你可以抢夺他身上的任何一件物品。

④攻击:当你与某个人位于同一位置时，你可以使用物品对其攻击。

5、同种物品伤害可叠加，若没有衣服，每件物品对你造成伤害+1，若你有衣服，你受到的伤害除以你的衣服数向下取整（结果可以为 0），斧头、拐杖、炮弹、雷的伤害为真实伤害，其造成伤害与衣服数量无关。

6、死者的所有物品归杀死者所有，被炮击倒的人所有物品都被销毁。

7、同一局中每种职业不超过两人，每个职业只能使用本职业的道具。

8、喷溅药水、炮弹为延时类攻击，当前回合使用后，下一回合结束后下回合开始前生效，施用位置所有人受到技能效果影响，喷溅药水与炮弹同时生效。

二、项目各模块与类设计细节

项目由 3 部分组成：网络通信、ui 框架、游戏规则。

1、网络通信部分：

由服务器端 Server 和客户端 Client 两部分组成。

服务器端 Server 负责监听端口，接收连接请求、广播数据。

客户端 Client 负责建立连接、发送和接收数据。

我们采用固定服务器的 IP 和端口的方式将每个游戏客户端连接到服务器上：打开服务器，服务器开始监听固定端口，客户端向服务器建立联系并开始传递消息。

因此，在开启软件前需要先打开服务器，否则会弹出弹窗——“错误”“无法连接服务器”。开启程序后，客户端自动与服务器连接，并开始信息的交互——连接状态、房间内玩家情况、数据交互。服务器负责处理房间——创建房间必须保证原有房间不存在，加入房间必须保证房间存在。游戏代码部分均在本地运行，服务器起到了收到数据后进行广播的作用。

这个模块的调试我们花费了非常非常长的时间：

a) 有些电脑相互连接不上

原因：在学校的 WiFi 网络中，由于存在隔离设置，同一个无线接入点（AP）之间的设备无法直接建立连接。

解决方法：我们经过大量实践发现了台式机与其他几台电脑的 ap 不同，所以将这台台式机设置为了固定的服务器用于处理数据的吞吐和协调游戏系统。

b) 信息莫名出现叠加在一起的情况。

原因：在网络波动的影响下，服务器接收不同数据，由于共用同一个写入函数 write，会出现内容在缓冲区的淤积的情况，导致内容的裁剪与拼接的发生。

解决方法：按照人物编号发送数据，即当 1 号玩家发送数据完毕后，2 号玩家才能发送，以此类推。并且每个信息前都加上特殊符号，通过特殊符号来隔断多个数据同时传输导致的拼接。

c) 网络波动导致断开连接

2、Ui 框架部分

打开游戏，此时界面有开始游戏、游戏规则和退出选项。我们选择开始游戏并且输入完昵称后，接下来会跳转到创建房间、加入房间。房间实时显示房间内成员的昵称，并且能确定谁是房间的所有者（房主）。房主拥有开始游戏的权力，只有房主可以启动游戏。接下来在房间内的每位玩家都会被要求选择一个角色。一旦每个玩家都选择完角色并确认准备就绪，游戏进入正式开始阶段。游戏按照回合制进行。每回合开始时，所有存活玩家的屏幕将显示剪刀石头布的按钮，以便玩家选择他们的动作。等待所有玩家选择完毕后，系统将根据玩家的选择进行判定，例如剪刀胜于纸，纸胜于石头，石头胜于剪刀。在判定完所有玩家的选择后，游戏将根据随机顺序让具有行动能力的玩家进行行动。每位玩家依次执行他们的行动，例如移动、攻击或使用技能等。与服务器通信更新信息：每次玩家完成行动后，游戏客户端将与服务器进行通信，更新每个玩家屏幕上的信息，例如角色位置、生命值、能量等。这确保了游戏状态的同步，并使所有玩家能够看到最新的游戏情况。

这个模块的调试我们也花费了很长的时间：

a) 在等待玩家点击按钮时没法进行其他运算。

原因：当前代码结构无法实现并行计算。

解决方法：将按钮的显示作为程序的断点，将原代码中的按钮显示（包括循

环中的按钮显示) 拆开, 只有点击上一个按钮, 才能执行下一个按钮及其相关运算。

b) 不同电脑结果不同

原因: 不同电脑随机数生成方式不同。

解决方法: 我们使用手写 mt19937、random_shuffle 等来实现每台电脑对于需随机内容处理的统一。

3、游戏规则部分

position 类用来储存不同位置下地雷, 药水, 炮弹的数量等。

player 类用来储存每位玩家的名字, 血量, 职业, 位置, 所属队伍, 存活状态, 装备栏, 购买权栏等。

game 类用来模拟整局游戏, 储存游戏人数, 玩家, 地图信息, 行动列表等。

where_to_move 函数用来得到可移动的位置

move/move_plus 函数用来实现玩家移动

game_start/next 函数用来模拟游戏的进行

who_to_touch/who_to_touch_plus 函数用来得到玩家能够接触的其他玩家

rob 函数用来实现抢劫功能

buy 函数用来实现购买功能

check 函数用来更新人物的死亡状态

attack 函数用来实现攻击功能

round_over 用来完成回合结束后状态的更新

game_over 用来判断游戏是否结束

click_+*用来模拟各种按钮按下后的效果

game_start_single 用来启动单人模式

game_start_team 用来启动组队模式 (为游戏简洁性暂未开放)

send 用来发送数据至服务器

receive 用来接收服务器发送的数据

show_button_+*用来显示某按钮

hide_button_+*用来隐藏某按钮

show_highlight_+*用来显示某高亮

hide_highlight_+*用来隐藏某高亮

`print_winner` 为打印胜者

`print` 为打印状态栏

三、小组成员分工情况

王晟帆：负责网络通信部分，代码 debug，美术设计，整理讨论结果。

陈益磊：负责 ui 界面设计与控制台书写，代码 debug。

林思旭：负责游戏本体代码书写，代码 debug，提供游戏创意，完善规则。

四、项目反思与总结

团队成员之间的合作非常重要。前期由于对 qt 的不熟悉，导致我们分工不是很理想——原来以为能很快解决的网络通信环节结果花掉了我们最多的时间与精力，尤其由网络通信环节导致整合时的各种奇奇怪怪的 bug。不过好在我们及时调整策略，先将 ui 部分与游戏本体组合，将网络通信的整合放到最后一步。尽管如此，bug 还是残存，花费了我们大量时间与精力进行修改。团队成员之间的有效沟通和正确分工是项目成功的关键。

在开发过程中，我们遇到了一些技术挑战，特别是在网络通信和 UI 框架方面。通过不断的调试和改进，我们成功解决了一些问题，例如处理不同电脑之间的连接问题、信息叠加和断开连接的处理等。这些经验使我们在后续的开发中更加成熟和高效。

游戏规则的设计是一个重要的环节。我们经过讨论和完善，确保游戏规则的合理性和平衡性。通过对职业、装备和行动的设定，我们为玩家提供了多样化的策略选择，使游戏更具挑战性和乐趣。此外，在软件的界面设计和交互方面，我们努力提供简洁、直观、易于操作的界面，使得用户可以在第一次接触该游戏时就明白操作逻辑，大大提升了用户的游戏体验。同时，我们也注重游戏的视觉效果和美术设计，使游戏画面更加精美和吸引人。

总的来说，开发《买刀买马》是一个充满挑战和成长的过程。通过团队的努力和协作，我们成功地学会了用 qt 实现游戏的开发，从中学到了许多宝贵的经验，并且相信这将对我们的项目开发和团队合作有所裨益。