



- 1.游戏玩法介绍
- 2.游戏代码介绍
- 3. 过程中遇到的问题以及解决措施
- 4. 后期版本优化期望

1.游戏玩法介绍

该游戏以飞机大战为模板,玩家使用鼠标移动丁老头,老头的头发本身会随时间增加不断掉落,同时界面上方随机出现四种事件图片,并以一定速度向丁老头撞去,相遇后老头头发数量发生变化,其中"熬夜"和"敲代码"会让老头头发变少而,"霸王生发液"和"植发"会让老头头发变多。老头的起始发量为3,左上角hair num显示动态变化的头发数量。

游戏界面展示



四种事件展示



熬夜: 熬夜伤肝掉头发

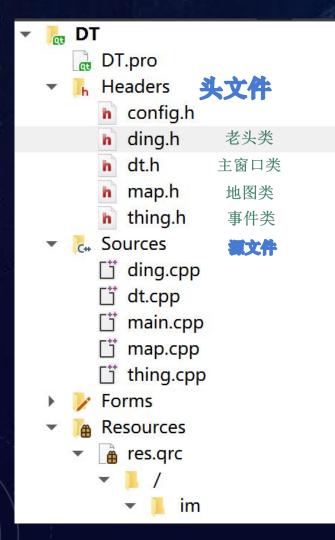








2.游戏代码展示





<config.h>(配置)

```
#ifndef CONFIG_H
    #define CONFIG_H
    #define GAME WIDTH
    #define GAME HEIGHT 1250
    #define GAME TITLE "丁老头脱发1.0"
    #define GAME RATE 10 //定时器单位毫秒
    #define MAP_SPEED 2 //地图滚动速度
9
10
11
    #define THING SPEED 10000;//东西移动速度
    #define THING_NUM 20;//总数量
12
13
    #define THING INTERVAL 50;//出场时间间隔
14
15
    #define IFSTART 0;
16
    #include<iostream>
17
18
    #endif // CONFIG H
19
```

Main函数 (执行主体)

```
#include "dt.h"
     #include <QApplication>
     #include<QResource>
     #include<QWidget>
     #include<iostream>
     #include"config.h"
     #include<QPushButton>
    int main(int argc, char *argv[])
10
         QApplication a(argc, argv);
12
          DT w;
13
         QResource::registerResource(":/im");
14
15
16
         w.show();
17
         return a.exec();
18
19
```

3.过程中遇到的问题以及解决措施

- 每隔一段时间掉 头发怎么控制?
- 定时器事件

```
void DT::playGame()
   //启动定时器
   m_Timer.start();
   //槽监视信号
   connect(&m_Timer, &QTimer::timeout,[=](){
      //东西出场
      thingtoscene();
      //更新游戏中元素坐标
      updatePosition();
      m_{ding.num-=0.01};
      update();
      //碰撞检测
      detect();
   });
             //显示num
             painter.drawRect(20,20,200,50);//画矩形框框
             QFont font("Courier",30);
                 painter.setFont(font);
                 painter.setPen(Qt::black);
                 painter.setBrush(Qt::red);
                 painter.drawText(20,50,"hair num:");
                 painter.drawText(165,50,QString::number((int)m_ding.num));
```

四种thing, 设置4个类?

```
© CSDN
```

```
//东西资源对象
QPixmap m_thing;

//位置
int m_x;
int m_y;

//东西的矩形边框(碰撞检测)
QRect m_Rect;

//状态
bool m_free;

//速度
int m_speed;

//更新时能否改变
```

bool m_change;

class thing

thing();

//类型设定

//更新坐标

void settype(int n);

void updatePosition();

//类型 int_type;

public:

```
thing::thing()
  //默认无参构造时
    type=0;
    m_thing.load(":/im/bawang1.png");
void thing::settype(int n)
   switch (n)
   case 0:
       type=0;
       m_thing.load(":/im/bawang1.png");
       break:
   case 1:
       type=1;
       m_thing.load(":/im/zhifa1.png");
       break;
   case 2:
       type=2;
       m_thing.load(":/im/daimal.png");
       break;
   case 3:
       type=3;m_thing.load(":/im/aoye1.png");
       break;
   default:
       break;
```

什么时候用THING::SET TYPE()函数?

```
void DT::thingtoscene()
    //出场时先设置type
    for(int i=0;i<20;i++)</pre>
       if(ts[i].m_change==1)
            int n=rand()%(4);
            ts[i].settype(n);
            ts[i].m_change=0;//下次更新时不能变
```

- 空闲vs非空闲?
- 绝对不能改非空闲的!
- 空闲态即为可改态(m_change=1)
- 出场那一瞬间更改(随机数)

```
void DT::detect()
{
    //遍历所有非空闲thing
    for(int i=0;i<20;i++)
    {
        if(ts[i].m_free)
        {
            continue;
            //空闲则执行下一次循环
            ts[i].m_change=1;
        }</pre>
```

不同类型的THING的不同碰撞效果如何实现?

• 碰撞效果? Hair num++/--

```
//相交, thing设置为空闲
if(ts[i].m_Rect.intersects(m_ding.m_Rect))
{
    ts[i].m_free=true;
    if(ts[i].type==0||ts[i].type==1)
        m_ding.ding_add();
    else
        m_ding.ding_minus();
}
```

```
class DING
public:
   DING();
   //设置老头位置
   void setPosition(int x,int y);
   //老头资源对象
   QPixmap d;
   //老头坐标
   int m_x;
   int m_y;
   //老头矩形边框
   QRect m_Rect;
   //老头头发数量
   float num;
    //头发数量变化
   void ding_add();
   void ding_minus();
   //头发增/减
   int state;//(0,1,2)对应(不变,增,减)
```

碰撞效果动态体现?给老头画头发

• 对DING::ding_add(),DING::ding_minus() 函数再优化->

什么时候画? DING类中引入int state变量,碰撞时 state改变

```
void DING::ding_add()
    num++;
    state=1;//该+
void DING::ding_minus()
    num--;
    state=2;//该-
```

怎么画?

#include<QPainter>

```
void DT::paintEvent(QPaintEvent *)
   QPainter painter(this);
   //绘制地图
   painter.drawPixmap(0,m_map.y1,m_
   painter.drawPixmap(0,m_map.y2,m_
  //绘制丁老头
   painter.drawPixmap(m_ding.m_x,m_
   //绘制thing
```

```
//给老头画头发
if(m_ding.state==1)
    painter.setPen(Qt::black);//设置画笔颜色
    int i=rand()%10;
                        随机生成斜度
    int j=rand()%10;
    painter.drawLine(m_ding.m_x-i,m_ding.m_y-j,m_ding.m_x,m_ding.m_y);
                                     画线
else if(m_ding.state==2)
    painter.setPen(Qt::black);//设置画笔颜色
```

BUT,掉头发时怎么擦除?

游戏开始界面按钮设置?

• 信号和槽

```
void DT::showbutton()
   //创建按钮
       QPushButton *but=new QPushButton("start",this);
   //调整位置
   but->move(250,700);
   //按钮上文字大小
   but->setStyleSheet("QPushButton{font:20px;}");
   //按钮尺寸
   but->resize(200,200);
     建立信息和槽 , 点击释放按钮之后, 将该按钮隐藏
   QObject::connect(but,&QPushButton::clicked,[=](){
       but->hide();
       ifrunbutton=0;
       playGame();
```

```
DT::DT(QWidget *parent)
    : QMainWindow(parent)
    ifrunbutton=1;
   m ding.state=0;
    //调用初始化场景
        initscene();
    if(ifrunbutton==1)
    showbutton();
```



```
DT::DT(QWidget *parent)
  : QMainWindow(parent)
  ifrunbutton=1;
  m_ding.state=0;
  //调用初始化场景
    initscene();
  if(ifrunbutton==1)
  showbutton();
  if(ifrunbutton==0)
  //启动游戏
  playGame();
```





按了之后没东西掉下来



大多数情况人比猴儿靠谱







这也不对



playgame



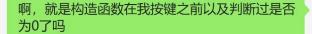
应该写到槽函数里



这样还是没用



因为构造函数执行的时候不符合条件就不执行那 句话了









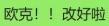


那应该写在槽函数里?





嗯





4. 后期版本优化期望

- (1.画面优化处理
- (2.随机数优化
- (3.头发增减动态体现
- (4.模式选择多样化

