Report-Angrybird   F74046551　凃登耀

MainWindow

-scene: QGraphicsScene *
-world: b2world *
-itemList: QList<GameItem*>
-timer: QTimer
-items: QList<GameItem *>
-block: QList<Block *>
-enemy: QList<Enemy *>
-rope: QList<Rope *>
-ball: QList<Ball *>
-temp: Whitebird *
-tempb: Blackbird *
-revolutejoint: QList<b2RevoluteJoint*>
-ropejoint: QList<b2RopeJoint*>
-ropelength: int
-weldjoint: b2WeldJoint *
-lcd: QLCDNumber *
-button: QPushButton *
-exit: QPushButton *
-x,y: double
-joint: b2MouseJoint *
-callback: Mycallback

+<<explicit>> MainWindow(parent: QWidget* = 0): void
+~MainWindow()
+showEvent(QShowEvent *): void
+eventFilter(QObject *,event: QEvent*): void
+closeEvent(QCloseEvent *): void
+<<signals>>quitGame(): void
-<<slots>>tick(): void
-<<slots>>QUITSLOT(): void
-<<slots>>restart(): void

---

b2ContactListener

mycontactlistener

-iscontacted: bool = 0

+BeginContact(contact: b2Contact *): void
+PreSolve(contact: b2Contact *, oldMainfold: const b2Manifold *): void
+PostSolve(contact: b2Contact *, impulse: const b2ContactImpulse *): void
+EndContact(contact: b2Contact *): void

Enemy

-health: int
-shoulddisappear: bool = 0

+Enemy(x: float , y: float , radius: float , timer: QTimer *, pixmap: QPixmap , world: b2World *, scene: QGraphicsScene *)
+setpixmap(pixmap: QPixmap , radius: float): void
+gethealth(): int&
+getshoulddisappear(): bool
+disappear(): void
+resetdisappear(): void

Land

+Land(x: float, y: float , w: float , h: float , pixmap: QPixmap , world: b2World *, scene: QGraphicsScene *)

Rope

+Rope(x: float , y: float, w: float , h: float , timer: QTimer *, pixmap: QPixmap , world: b2World *, scene: QGraphicsScene *)

Ball

+Ball(x: float, y: float , radius: float , timer: QTimer *, pixmap: QPixmap, world: b2World *, scene: QGraphicsScene *)

GameItem

#g_body: b2Body*
#g_pixmap: QGraphicsPixmapItem
#g_size: QSizeF
#g_world: b2World *
#g_worldsize, g_windowsize: static QSizeF
#isUsecollide: bool = 0

+GameItem(world: b2World *)
+~GameItem(): virtual
+setGlobalSize(worldsize: QSizeF, windowsize: QSizeF): static void
+getBody(): b2Body&
+usecollide(): void
+getcollide(): bool
+resetcollide(): void
+<<slots>>paint(): virtual void
+<<slots>>collide(): virtual void

Bird

+Bird(x: float , y: float , radius: float , timer: QTimer *, pixmap: QPixmap , world: b2World *, scene: QGraphicsScene *)
+setpixmap(pixmap: QPixmap , radius: float): void
+setLinearVelocity(velocity: b2Vec2): void
+setDensity(): void
+collide(): virtual void
+~Bird(): virtual

Bird

+Bird(x: float , y: float , radius: float , timer: QTimer *, pixmap: QPixmap , world: b2World *, scene: QGraphicsScene *)
+setpixmap(pixmap: QPixmap , radius: float): void
+setLinearVelocity(velocity: b2Vec2): void
+setDensity(): void
+collide(): virtual void
+~Bird(): virtual

Blackbird

-temptimer: QTimer *
-tempworld: b2World *
-tempscene: QGraphicsScene *
-tempbird: Bird*
-tempbird1: Bird*
-tempbird2: Bird*
-tempbird3: Bird*

+Blackbird(x: float , y: float , radius: float , timer: QTimer *, pixmap: QPixmap , world: b2World *, scene: QGraphicsScene *):
+collide(): virtual void
+~Blackbird(): virtual
+clear(): void

Yellowbird

+Yellowbird(x: float , y: float , radius: float , timer: QTimer *, pixmap: QPixmap , world: b2World *, scene: QGraphicsScene *)
+collide(): virtual void

Whitebird

+Whitebird(x: float , y: float , radius: float , timer: QTimer *, pixmap: QPixmap , world: b2World *, scene:QGraphicsScene *)
+collide(): virtual void

按下鼠標開始移動紅鳥(按鍵盤:無功能)

按下第 2 次鼠標時 黃鳥出動(按鍵盤:向下衝)

第 3 次按下鼠標 黑鳥出動(按鍵盤: 爆裂成 4 隻小黑鳥)

第 4 次按下鼠標　白鳥出動(按鍵盤: 變大)

最後:全部彈完後　按 Restart 重新開始

Box2d 引擎, 鳥、球和敵人是圓形 , 方塊和彈弓是方形(彈弓與物體忽略碰撞),
繩子由諸多細小小方塊實作