# Project 3 - Angry Bird

瞿旭民(Kevin Chiu)

徐朝逸(LuisHsu)

## 憤怒雞?





(懂得就拜託笑一下吧<(\_ \_ )> ....)

# Nope!是憤怒鳥la!



https://youtu.be/uypaLQ6UoRw?t=2m38s

### 好啦 開玩笑地~ 當然還是做原版的~

https://www.youtube.com/watch?v=aiiQ8btusrs

•說了那麼多,該進正題囉!

### 遊戲需求(AngryBirds Requirements)

- 鳥必須要有碰撞的反應
  - 反彈、障礙物反彈...
  - 可以接受障礙物碰到直接消失=>計分!
- 必須要有重力(整個視窗的物件都需要被重力的影響!)
- 必須實作3種不同的鳥(不包含原本的紅鳥)
- 必須要有障礙物
- 必須要有敵人!
- •當要離開時,必須emit "quitGame()"的signal
- => 所以, 必須強制使用多型與繼承來實作唷!



### Why?

因為你們每個人的遊戲, 到最後都會被串起來, 變成一個真正可以玩的憤怒鳥喔~



#### 評分標準

- Basic GUI(包含quitSignal()的結束按鈕, 以及 restart的部份): 10%
- Barrier and Enemy: 10%
- Birds(>=4) : 20%
- Physics : 10%
- Report : 10%
- Demo: 40 %
- •Bonus : up to 20 %!!

- Deadline : 6/19
- Demo: 6/20 ~ 24(記得要填時間!!)

記得提早開始寫作業!!!!

#### 物理引擎?

- •由於這個遊戲內容比較需要模擬動態的環境,所以免不了的,一定會需要物理引擎。在這個project中,我們引入了Box2D作為輔助工具,讓我們的遊戲看起來更加流暢!
- Box2D 官方網站: http://box2d.org/
- 共筆頁面(提問區): https://goo.gl/qDN7zc

• 當然, 如果你覺得你有更好的方式, 也可以使用!



#### Demo

· 當然, 光說不做不是好事, 讓同學看看用box2D做出來的效果!

如果想挑戰box2D的同學,可以有以下方案選擇: 助教們會提供各位一個簡單的範例code,如果想要的話,可以從範例 code修改!(而且已經幫各位弄好box2D的lib囉,分別提供linux與 windows版本!)

Example Code on github: <a href="https://github.com/ncku-pd2/Example-code-for-Project-3">https://github.com/ncku-pd2/Example-code-for-Project-3</a>

#### UML Diagram

```
MainWindow
-scene: QGraphicsScene *
-world: b2World *
-itemList : QList<GameItem *>
-timer: QTimer
                                                                                                   Gameltem
+<<explicit>> MainWindow(parent:QWidget *=0): void
                                                                         #g body: b2Body *
+~MainWindow()
                                                                         #g pixmap: QGraphicsPixmapItem
+showEvent(OShowEvent*): void
                                                                         #g size: QSizeF
+eventFilter(QObject *,event:QEvent): bool
                                                                         #g world: b2World *
+closeEvent(OCloseEvent*): void

⇒ #g worldsize: static QSizeF
+<<signals>> quitGame(): void
                                                                         #g windowsize: static QSizeF
-<<slots>> tick(): void
                                                                         +GameItem(world:b2World *)
-<<slots>> QUITSLOT(): void
                                                                         +~GameItem()
                                                                         +setGlobalSize(worldsize:QSizeF,windowsize:QSizeF): static void
                                                                         +<<slots>> paint(): void
                                                                   Bird
                                                                                                                                    Land
                                        +Bird(x:float,y:float,radius:float,timer:QTimer *,
                                                                                                            +Land(x:float,y:float,w:float,h:float,pixmap:QPixmap
                                                                                                                  world:b2World *,scene:QGraphicsScene *)
                                              pixmap:QPixmap,world:b2World *,scene:QGraphicsScene
```

+setLinearVelocity(velocity:b2Vec2): void

#### Hint

- 場景控制
  - QGraphicsView
  - QGraphicsScene
  - QGraphicsItem
- 物理引擎
  - object's attribute (hand-made)
  - box2D physics engine
  - QPropertyAnimation Framework (State machine)

#### Ask Question

Send E-mail: pd2@imslab.org

鑑於moodle簡訊需要等待轉寄上的問題,大家如果有問題,就寄到這個信箱裡頭吧!(No Moodle message to TA)

Or ... Ask TA!