太鼓の達人 project H34041092\_陳逸帆

一、遊玩方式

在建置前，因為此程式需要讀取外部的歌曲和譜面，請先到file\_control.h內的path更改絕對位置到譜面所在的資料夾。注意，資料夾為歌名，資料夾內需有一個以歌名為檔名的.tja檔(即為太鼓次郎譜面，但因為本程式無法讀取Shift\_JIS、EUC-JP，請先改為unicode編碼)。

然後就可以開始了!

首先是開始畫面



在此畫面按D和K鍵會有ka的聲音按F及J鍵會有dong的聲音和開場音樂並進到選歌畫面



這是選歌畫面，D是上一首歌、K是夏一首歌，F和J是選取

然後是設定和選難度畫面



這裡有三種設定和四種難度可以選，設定選取後底部會變。

ランダム 是隨機，譜面會隨機生成

30秒モード 是倒數30秒模式

オート 是自動模式，系統會自動幫你打

這裡先示範全開的困難



最左上是小咚，他會隨節奏上下動並在每10combo時往上跳，良、可、不可時分別吐出不同的魂。

中間上面的是30秒到數，左上角是歌名。

左邊中間道右依序是難度，分數，combo，判定。

下面是5個舞者，會在分數夠的時候分別出來跳舞並在miss過多的時候回去

如:



接下來是成績結算，會先看到一個背景變換然後會依據你的正確率決定過關與否





UML

類別名稱:

notes

資料成員

+symbol:int

類別名稱:

file\_conrtol

資料成員:

+dir : Qdir

+path : QString

+map : QMap<QString,float>

+songMap : QStringList

+TITLE : QString

+SUBTITLE : QString

+,WAVE : QString

+trackPath : QString

+star[4] : int

成員涵數:

songlist\_init() : void

setSongPlay(QString) : void

getStar(QString) : void

getMap(QString,int) : void

getSubtitle(QString) : QString

convert(bool) : void

類別名稱:

MainWindow

資料成員:

+scene : Scene

成員涵數:

settingBg() : void

類別名稱:

Scene

資料成員:

QString song; QSound \*op, \*dong, \*ka; QGraphicsPixmapItem \*a,\*b,\*c; int screenMode=0;//0=start 1=pick music 2=setting //3=playing 4=after play //picksong int songs=0,s=1; QGraphicsSimpleTextItem \*text0,\*text1,\*text2; QList<QString> title,subtitle; //setting QGraphicsPixmapItem \*d,\*st,\*picked\_background\_r,\*picked\_background\_3,\*picked\_background\_a; QGraphicsPixmapItem \*arrow,\*icon\_rand,\*icon\_30sec,\*icon\_auto; int now\_on=0; int difficulty=1;// bool setRandom=false; bool setHidden=false; bool setAuto=false; bool set30secMode=false; //play QGraphicsPixmapItem \*e,\*randomTag,\*countdownTag,\*autoTag; file\_control Map; QMediaPlayer \*track; QList<QSound \*> comTrack; int VOL=70,p=0; QTimer \*timer,\*timer1,\*timer2,\*timer3,\*countdown; QElapsedTimer \*Etimer; QList<notes \*> list; //QList<QGraphicsPixmapItem \*> numbers; QStringList songMap; int great=0,good=0,miss=0,score=0,combo=0,maxcombo=0; int soul=0;//48pass 63full int missCombo=0,rate=0; int firstlayer=1,secondlayer=0; float BPM; int msec,test; int waitfor; int secLeft=30; //animation QTimer \*RD,\*RK,\*LD,\*LK; QGraphicsPixmapItem \*rdp,\*ldp,\*rkp,\*lkp; bool add[4]={false,false,false,false}; QTimer \*soundTimer,\*AniTimer1,\*AniTimer2,\*AniTimer3,\*AniTimer4; QList<QGraphicsPixmapItem \*> Judge; QList<QGraphicsPixmapItem \*> expl; QPixmap ex1; QPixmap numberPic[10]; QPixmap ctdnumberPic[10]; QPixmap scorenumberPic[10]; QList<QGraphicsPixmapItem \*> numbers,ctdnumbers,scorenumbers; QPixmap jumpPic[3]; int jumpPicNum=1; QGraphicsPixmapItem \*jumper; QList<QGraphicsPixmapItem \*> FSlist,MSlist,WSlist; int dancer\_ctr[6][2]={0},toNum=0,danceNum=0,dscore=500; QGraphicsPixmapItem \*d1,\*d2,\*d3,\*d4,\*d5; QPixmap d1pa[16],d1pd[16],d1pn[16],d2pa[16],d2pd[16],d2pn[16],d3pa[16],d3pd[16],d3pn[16]; QPixmap d4pa[16],d4pd[16],d4pn[16],d5pa[16],d5pd[16],d5pn[16]; //result QSound \*seiseki,\*fullcombo; bool passed=true;

成員涵數:

void keyPressEvent(QKeyEvent \*event) : void mousePressEvent(QGraphicsSceneMouseEvent \*event) : void

setbg\_0() : void

setbg\_1() : void

unsetbg\_1() : void

setbg\_2() : void

unsetbg\_2() : void

setbg\_3() : void

unsetbg\_3() : void

setbg\_4() : void

unsetbg\_4() : void

showscene1() : void

scene1left() : void

scene1right() : void

nowOn() : void checked() : void

loadDancerPic() : void

createDanceItem() : void

removeDanceItem() : void

setMap() : void

startPlay() : void

eliminate(int) : void

playSeiseki() : void

Generate() : void

Move() : void

Auto() : void

startWave() : void

stopWait() : void

endPlay() : void

secondsLeft() : void

playRdong() : void

playLdong() : void

playRka() : void

playLka() : void

GenGreat() : void

GenGood() : void

GenMiss() : void

moveJudgement() : void

ShowCombo() : void

GenExplode() : void

RemoveExplode() : void

ShowScore() : void

Jumper() : void

Dance() : void

donJump() : void

FlySoul() : void

MidSoul() : void

WalkSoul() : void

GenFlySoul() : void

GenMidSoul() : void

GenWalkSoul() : void

PlayRdong() : void

PlayLdong() : void

PlayRka() : void

PlayLka() : void

Great() : void

Good() : void

Miss() : void

Explode() : void

Combo() : void

Score() : void

Jump() : void

SoulF() : void

SoulM() : void

SoulW() : void

程式架構

本程式主要由一個讀檔的class和一個用來操控遊戲畫面的程式

Class File\_control主要使用QDir、QFile和QStringList

先以QDir看有哪些資料夾再以QFile開檔、讀檔並存入QStringList。

Class Scene則是負責所有的遊戲控制，其中主要使用connect功能將時間與其他函數連結起來以達到動畫、譜面生成等目的。