

# 國立虎尾科技大學

## 機械設計工程系

### 計算機程式 ag6 期末報告

#### PyQt5 事件導向計算器

PyQt5 Event-Driven Calculator Project-under 40623134

學生：

設計一甲 40623134 紀冠瑋

設計一甲 40623125 鍾旻諺

設計一甲 40623127 張育偉

設計一甲 40623135 洪銘琪

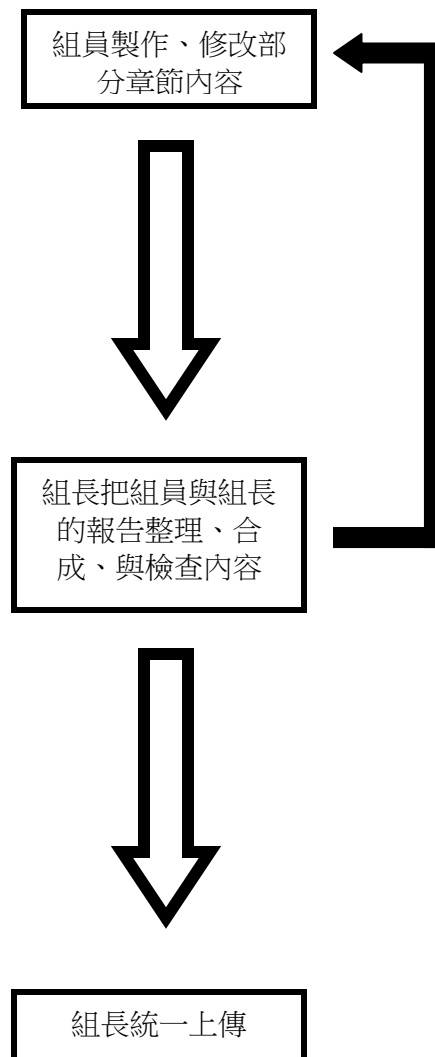
設計一甲 40623136 黃子軒

設計一甲 40623145 林暉恩

指導教授:嚴家銘

2017.12.21

我們是利用 word 製作這次的報告的流程如下：



# 目錄

圖目錄 .....	iii
第一章 前言 .....	1
第二章 可攜程式系統介紹 .....	2
2.1 啟動與關閉 .....	2
第三章 Python 程式語法 .....	3
3.1 變數命名 .....	3
3.2 print 函式 .....	3
3.3 重複迴圈 .....	4
3.4 判斷式 .....	4
3.5 數列 .....	5
第四章 PyQt5 簡介 .....	6
4.1 PyQt5 架構 .....	6
4.2 建立按鈕 .....	7
第五章 Calculator 程式 .....	8
5.1 建立對話框 .....	8
5.2 建立程式碼 .....	8
第六章 心得 .....	9
6.1 Github 同倉儲 .....	9
6.2 學員心得 .....	9

第七章 結論 .....	10
7.1 結論與建議 .....	10

## 圖目錄

圖 4.1 .....	6
圖 4.2 .....	7
圖 4.3 .....	7
圖 5.1 .....	8
圖 5.2 .....	8
圖 5.3 .....	8
圖 5.4 .....	8
圖 5.5 .....	8
圖 5.6 .....	8

# 第一章 前言

前言內容。

第一次一夥人寫程式，對於剛接觸到程式邊緣的我們，老實說大家都想抱大腿，不過很幸運我們沒有那種機會，所以我們在製作計算機時比較懂得語法要做甚麼，只是不清楚指令上的用法。

## 第二章可攜程式系統介紹

### 2.1 啟動與關閉

- 按下 start.bat 會開啟 cover and abstract 與 cmd.exe 。
- 按下 stop.bat 會關閉所有 cover and avstract 與 cmd.exe 。
- 從 cover and abstract 中修改 start.bat 的程式。
- 開啟時並開啟 leo.py 程式。
- 修改 leo.py 中的程式，程式會自動收尋要開啟的檔案。

### 2.2 可攜式系統

- 提供電腦編輯環境讓自己能夠隨身隨的完成工作和開發程式
- GIMP2：用以裁切圖片
- DiaPortable：用以繪製圖形與註解圖片。
- github：網際合作管理，能多人協同，分工合作。
- Python：在電腦中能進行 Python 許多程式開發，編譯出來的.py 檔功能強大。
- Share-X：擷取圖片與錄製影片，能自動上傳。

## 第三章 Python 程式語法

### Python 程式語法

#### 3.1 變數命名

1. 通用命名規則: 函式命名，變數命名，文件命名要有描述性；少用縮寫，盡可能地使用描述性的字元來呈現，方便其他人來觀看，以及易於檢查。
2. 文件命名: 文件名要全部小寫，可以包含底線( )或連字符(-)。如果有專案約定的情況，則照專案約定來使用，如果沒有專案約定( )會更好。
3. 變數命名: 變數名一律小寫，單詞之間用底線連接。類別的成員變量以下劃線結尾，但結構體的就不用。

#### 3.2 print 函式

函式為結構化程式，將相同功能的程式獨立出來，經由函式的呼叫，傳入資料與回傳處理後的結果，只要將函式寫好，可以不斷利用此函式做相同動作，可以達成程式碼不重複，要修改此功能，只要更改此函式。

**函式的定義:** 以 `def` 開頭，空一個空白字元，接函式名稱後，串接著一對小括號，小括號可以填入要傳入函式的參數，當參數有多個的時候以逗號隔開，右小括號後面須接上「:」，函式範圍為縮行固定個數空白字元的程式碼，縮行相同空白字元的程式碼就是函式的作用範圍。當函式需要傳回值使用指令 `return`，表示函式回傳資料給原呼叫函式，若不需要回傳值的函式就不需要加上 `return`。



### 3.3 重複迴圈

- 1.處理重複事件的敘述稱為迴圈，有[while]、[for]、[do...while]。
- 2.[for]是屬於固定次數的迴圈，[while]為不固定次數的迴圈。
- 3.[for]、[while]為測試在執行，屬於前測試迴圈；[do...while]為執行後在測試，屬於後測試迴圈。

### 3.4 判斷式

- 1.定義:在程式中判別是否成立要或不成立的一種語法。
- 2.判斷的條件:

比較運算 (A、B 比較)	語法
等於	$A==B$
不等於	$A!=B$
大於	$A>B$
小於	$A<B$
不大於	$A\leq B$
不小於	$A\geq B$

3. if-else 的使用 If 的條件:成立時通過成立時要跑的程式。 Else 的條件:不成立時通過不成立要跑的程式。

### 3.5 數列

簡單的解釋就是說我把所需要的資源全部放在一塊，然後讓我在後續的工作不需要再去一個一個找，而是直接重我所放資源的地方找。

## 第四章 PyQt5 簡介

說明 PyQt5 基本架構與程式開發流程

### 4.1 PyQt5 架構

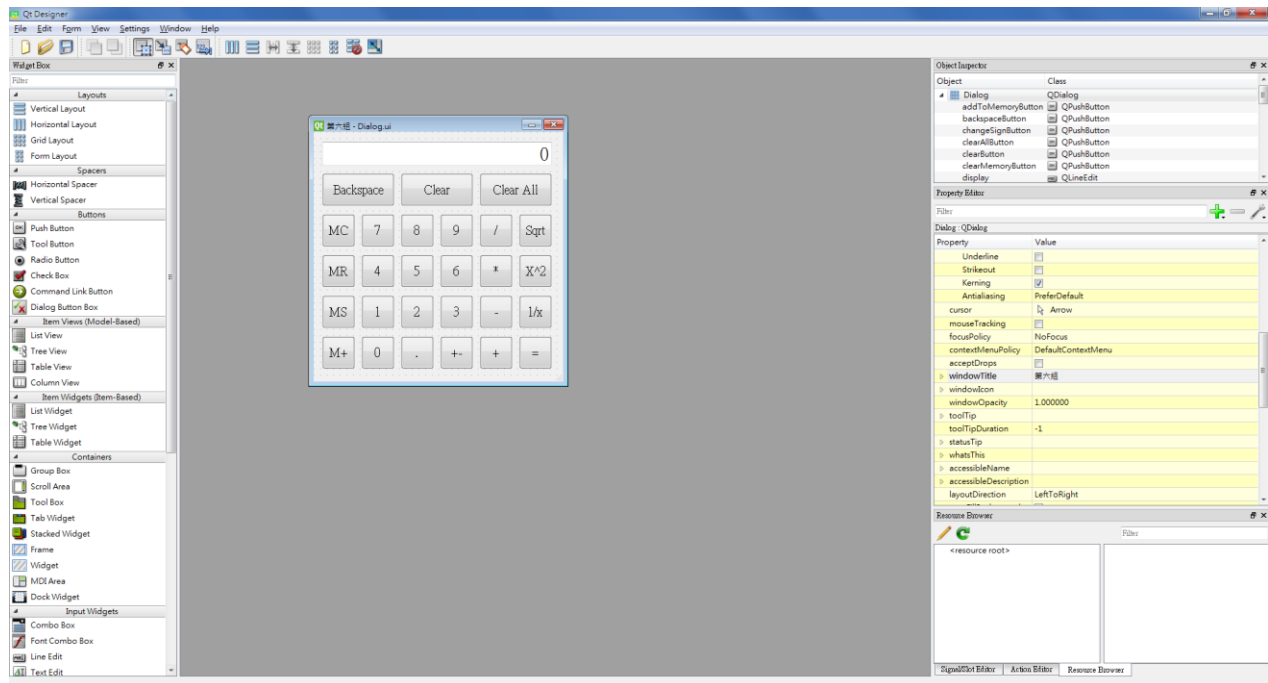


圖 4-1

## 4.2 建立按鈕

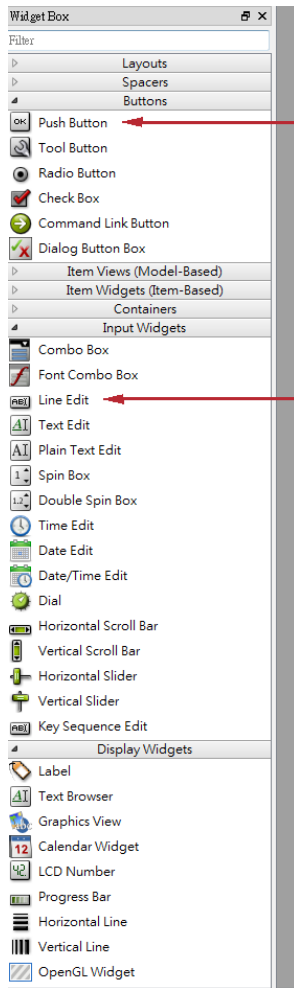


圖 4-2

可壓按鈕

顯示條

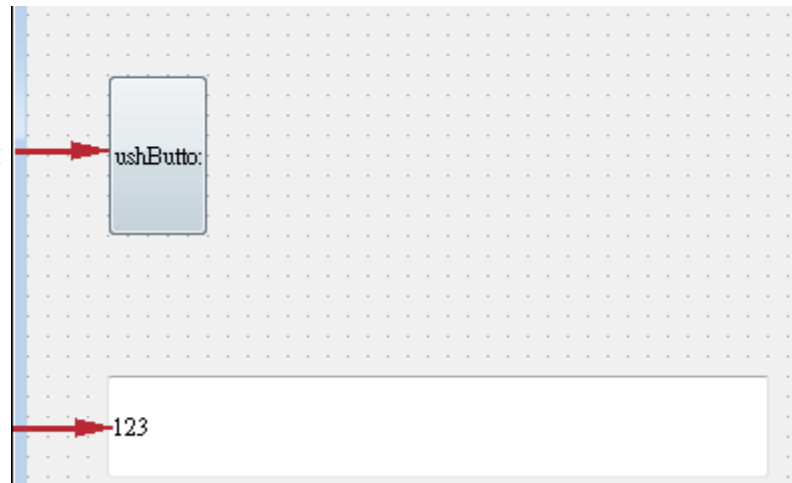


圖 4-3

## 第五章 eric6 程式

## 5.1 建立對話框

先創建一個 project，再打上圖片上的資料，在 Forms 新建 New form 之後會跳出一個視窗點選 OK 後建立 ui 資料夾，名字輸入 Dialog，對話框就新建好了。

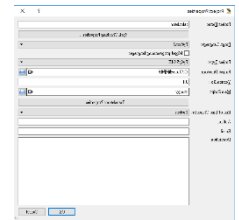


圖 5-1



## 5.2 建立程式碼

在 run.py 上輸入：

```
import sys
from PyQt5 import QtWidgets
from ui.Dialog import Dialog

app = QtWidgets.QApplication(sys.argv)
win = Dialog()
win.show()
sys.exit(app.exec())
```

圖 5-2

圖 5-3

## 再到 ui 裡輸入程式碼

```

42 self.equalButton.clicked.connect(self.equalClicked)
43 @property
44 self.backspaceButton.clicked.connect(self.backspaceClicked)
45 @property
46 self.changeDisplayButton.clicked.connect(self.changeDisplayClicked)
47 @property
48 unaryOperator = [self.equalButton.toggled, self.powerButton.toggled, self.reciprocal
49 for i in unaryOperators]
50 clickedOperator = self.unaryOperatorClicked
51 # 设置变量
52 self.setMemoryButton.clicked.connect(self.setMemory)
53 # 加上运算符
54 self.addMemoryButton.clicked.connect(self.addMemory)
55
56 def digitClicked(self):
57     使用数字按下时，0 应该放在数字前面
58     当前数字已经不为 0，再按下数字 0 时，应仍保持为 0 或 0.
59     0
60     button = self.sender()
61
62     if self.display.text() == "0" and self.display.text() == "0.":
63         return
64     # 当前数字 已经不为 0
65     if self.waitingForOperand:
66         self.display.clear()
67         self.waitingForOperand = False
68     # 添加数字
69     self.display.setText(self.display.text() + button.text())
70
71 def unaryOperatorClicked(self):
72     "一元运算符按下" 处理逻辑
73     clickedButton = self.sender()
74     clickedOperator = clickedButton.text()
75     operand = float(self.display.text())
76
77     if clickedOperator == "1/x":
78         if operand < 0.01:
79             self.alertOperation()

```

圖 5-4

[illegible]

圖 5-5

```

81         return
82
83     result = math.exp(operand)
84     self.clickedOperator = "="; result = result
85
86     self.clickedOperator = math.pow(operand, 2.0)
87
88     self.clickedOperator = "1/x";
89     if operand == 0.0:
90         self.abortOperator()
91         return
92
93     result = 1.0 / operand
94
95
96     self.display.setText(str(result))
97     self.waitingForOperand = True
98
99
100 def self.clickedOperatorClicked(self):
101     """按下等于按钮的回调函数"""
102     clickedButton = self.sender()
103     clickedOperator = clickedButton.text()
104     operand = float(self.display.text())
105     处理结果
106     if self.pendingMultiplicativeOperator:
107
108
109         计算：self.calculate(乘数或除数，操作数)
110         返回 bool 以指示是否应清除显示
111         Python 3 中 if not 语句的用法：要清除待执行的 self.abortOperator() +=
112
113         if not self.calculate(operand, self.pendingMultiplicativeOperator):
114             return
115         self.display.setText(str(self.factorSoFar))
116     清除 operand 并 self.factorSoFar
117     self.factorSoFar = self.factorSoFar * operand if operand != 0.0
118     self.pendingMultiplicativeOperator = None
119
120 清除结果
121 if self.pendingAdditiveOperator:
122
123     同上

```

圖 5-6

## 第六章心得

期末報告心得：

### Github 協同倉儲 6.4 學員心得

40623134:負責 0~9, +, 清除鈕。以前常用機算機，但自己編寫設計時感覺非常新鮮，能運作的時候很開心。

40623135: 這學期學習這門計算機程式課，對於我我認為是很挑戰自己的一門課程，對於老師所教得東西都是第一次接觸非常的陌生，但是經過老師的教學影片以及同學的幫助我盡力完成了老師所教的東西，雖然沒做的很好，但是學習到了程式的基礎。

40623125:負責+/-，還有修改一些其他的小錯誤，當自己完成的時候心裡非常的有成就感，沒想到自己平常在用的東西，有一天能夠靠自己上課學的東西完成其中的一部分，雖然原本對電腦、程式這方面沒甚麼興趣，但是希望經過這段時間的練習之後，能慢慢培養自己對電腦的興趣，因為以後出社會工作之後，電腦一定是不可或缺的技能條件。

40623127:修改 Bug，製作計算機起初很有趣，雖然之後逐漸深入讓課程有了難度，但是在組員及同學的幫助下已可以順利完成計算機。

40623136 黃子軒:

**Fossil SCM:** Fossil SCM 是一個非常方便的網路倉儲，不管有沒有網路都可以使用，只要帶著隨身碟，走到哪，把 USB 一插進去就可以繼續工作。

網誌心得:以前看到別人做的網誌，都只覺得很厲害。但自己開始製作網誌時，真的很不容易。但學會一些後慢慢覺得也有有趣的地方。

**Github 協同倉儲:**github 協同與同組的一起做出計算機。計算機一開始真的甚麼都聽不懂，還要跟同組的協同，真的很傷腦筋。而且同組的上傳後還會出現 bug，中間雖然出現了很多問題，但慢慢一一解決，最後還是把計算機做出來了。

學員心得:計算機程式這門課要學的好真的很不容易，不管是從剛開始的 fossil 還是 github。但慢慢學卻也學到了很多東西，像是如何建立自己的倉儲亦或是建立計算機，還有要與同組解決 bug。這門課使我獲益良多，之後也會更加努力學習。

## 第七章結論

### 7.1 結論與建議

學期末了，到了驗收整個學期成果的時期，從起初剛接觸寫程式的世界，fossil、透過 PyQt5 calcuator 程式原始碼學 Python3 語法、製作 PyQt5 計算機程式，雖然有些人之前有學過，但是對大多數的人來說卻是新的領域，上課完總是一頭霧水只能求助於同學或觀看老師上課錄製的影片，雖然過程中有困難有挫折，但現在已能和組員們順利的在時限內完成老師所指示的作業。