

第1次練習-練習-PC1

學號：112111208

姓名：黃誌麒

作業撰寫時間：大約25 (mins · 包含程式撰寫時間)

最後撰寫文件日期：2024/10/20

本份文件包含以下主題：(至少需下面兩項，若是有多者可以自行新增)

- ☒ 說明內容
- ☒ 個人認為完成作業須具備觀念

說明程式與內容

開始寫說明，該說明需說明想法，並於之後再對上述想法的每一部分將程式進一步進行展現，若需引用程式區則使用下面方法，若為.cs檔內程式除了於敘述中需註明檔案名稱外，還需使用語法```語言種類 程式碼```，其中語言種類若是要用python則使用py，java則使用java，C/C++則使用cpp，下段程式碼為語言種類選擇csharp使用後結果：

```
public void mt_getResult(){  
    ...  
}
```

若要於內文中標示部分網頁檔，則使用以下標籤```html 程式碼```，下段程式碼則為使用後結果：

```
<%@ Page Language="C#" AutoEventWireup="true" ...>  
  
<!DOCTYPE html>  
  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head runat="server">  
<meta http-equiv="Content-Type" ...>  
    <title></title>  
</head>  
<body>  
    <form id="form1" runat="server">  
        <div>  
            </div>  
    </form>  
</body>  
</html>
```

更多markdown方法可參閱<https://ithelp.ithome.com.tw/articles/10203758>

請在撰寫"說明程式與內容"該塊內容，請把原該塊內上述敘述刪除，該塊上述內容只是用來指引該怎麼撰寫內容。

1. 請執行下面Git操作 (請參照題目pdf)

Ans:請看圖檔answer.png

2. 請整理課堂上所提的Git指令，顯示其語法語給予明顯的例子

Ans:

add(加入)

例子:創一個檔名為為ty的檔案，並在終端輸入`git add ty`

結果:會將ty這個檔案加入至暫存區

commit(提交)

例子:在終端輸入`git commit`

結果:將暫存區的ty檔案提交至客戶端的倉庫

push(推送)

例子:在終端輸入`git push`

結果:將提交的數據從客戶端推送到伺服器端中

fetch(獲取)

例子:在終端輸入`git fetch`

結果:從伺服器端的倉庫分支獲取資料至客戶端倉庫

pull(拉取)

例子:在終端輸入`git pull`

結果:將伺服器新的倉庫分支與客戶端新的倉庫分支合併(fetch+merge)

branch(分支)

例子:在終端輸入`git branch new`

結果:建立名為new的分支

checkout(切換)

例子:在終端輸入`git checkout new`

結果:切換至分支new

merge(合併)

例子:在終端輸入`git merge new`

結果:將分支new合併至分支節點

3. 請先寫出同學認為資料結構一門課每周需要花多少時間？並提出理由。並將「請問要學習大學的資料結構每一週需要對於一般大學生需要多少時間」貼入chatgpt，並將其所建議的敘述貼出後，比較同學所認為若是每周低於chatgpt所評估的時間，兩者主要差異為何？應該怎麼安排自己時間了解資料結構？

Ans:

我認為一周大約花4小時，算是一個比較適中的時間，畢竟時間拉長可能會消化不良，太短又會學不到什麼東西。

chatgpt:學習大學的資料結構對於一般大學生來說，每週的學習時間可能會因人而異，但一般建議的

時間範圍是每週10到15小時。這包括了以下幾個方面：

上課時間：通常課堂會有2到3小時的講授。

閱讀教材：每週的閱讀和理解課本或其他資料，可能需要3到5小時。

作業和實作：編寫程式碼和完成作業，可能需要4到7小時，視作業難度而定。

複習和準備考試：定期的複習和考試準備也會占用一些時間。

如果你對資料結構有特別興趣或背景知識，可能會需要更少的時間；相反，如果你是初學者，可能需要更多的時間來消化和理解概念。最重要的是保持穩定的學習進度，定期練習，這樣會更有助於掌握資料結構的內容。

首先我的回答是沒有包含上課時間的，所以相加後就相當於是8個小時，至於跟GPT的回答還是有差異我認為是因為我比較沒有花時間在預習以及複習上，都是手動實作偏多。

我對於目前這樣的安排，其實覺得沒什麼不好，實作當下遇到問題就趕緊問，其實也是可以學到很多。

個人認為完成作業須具備觀念

開始寫說明，需要說明本次練習需學會那些觀念 (需寫成文章，需最少50字，並且文內不得有你、我、他三種文字)且必須提供完整與練習相關過程的notion筆記連結

本次練習需學會的觀念：

1. 了解git的基本指令，並在執行完後觀察git graph線圖的變化
2. 認真評估對於資料結構，以自己的能力需要花多少時間在上面