

Lab 1

注意事項：

1. Lab 的繳交期限為**星期二(2/21) 23:59**。
2. Lab 的分數分配：Lab 分數 100%，Bonus 20%。
3. 請儘量於 Lab 時段完成練習，完成後請找助教檢查
4. 檔名規定：
 - **Lab1&2** 請用 **學號_Lab number 為檔名(學號_Lab)** 做一個檔案夾 (e.g., N96091350_Lab1), 將你的全部 ipynb 檔放入檔案夾，**壓縮後上傳**至課程網站 (e.g., N96091350_Lab1.zip)。
 - **Bonus** 請用 **學號_bonus 為檔名(學號_bonus)**做一個檔案夾 (e.g., N96091350_bonus), 將你的全部 ipynb 檔放入檔案夾，**壓縮後上傳**至課程網站 (e.g., N96091350_bonus.zip)。

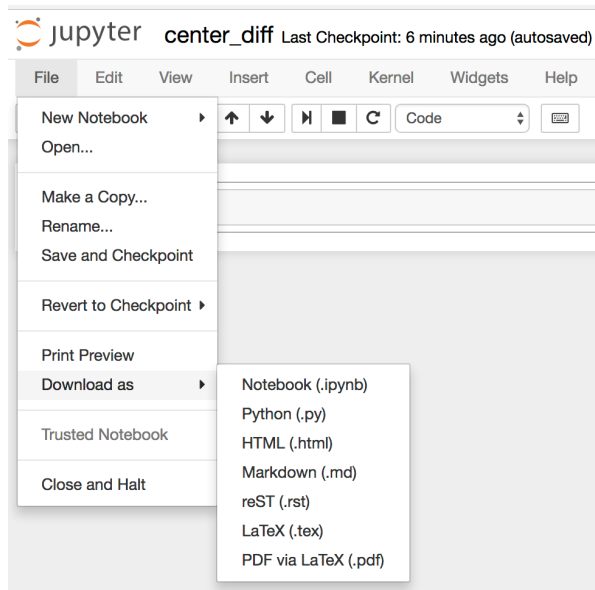
檔名錯誤將記為 0 分

5. 上傳後即可離開。
6. 未完成者可於下周一 (2/27) 09:00 am 前上傳至 Moodle，惟補交的分數將乘以 **0.8** 計，超過期限後不予補交。
7. Bonus 需於下周一 (2/27) 09:00 am 前上傳至 Moodle，不予補交。
8. 準時繳交者，請交至「Lab1 準時繳交區」；補交者，請交至「Lab1 補交區」；bonus 請繳交至「bonus 繳交區」

請勿抄襲，抄襲者與被抄襲者本次作業皆 0 分計算

Lab Submission Procedure (請仔細閱讀)

1. You should submit your Jupyter notebook and Python script (*.py, in Jupyter, click File, Download as, Python (*.py)).



2. Name a folder using your student id and lab number (e.g., n96081494_lab1), put all the python scripts into the folder and zip the folder (e.g., n96081494_lab1.zip).
3. Submit your lab directly through the course website.

Total 120%

1. **(50%)** Name your file `RockPaperScissors.ipynb`. Write a program to play Rock, Paper, Scissors game. The game will (1) ask you to enter Rock, Paper, or Scissors, (2) randomly generates Rock, Paper, or Scissors and (3) determine whether it is a win, tie or loss. The game will quit when you hit the first win. (**hint**: you can import the random module from python by calling `import random` and use `random.randint(1, 3)` to generate a random number 1, 2, or 3).

Below are two sample runs:

(Sample Run 1, **bold** is input from keyboard)

```
Welcome to ROCK, PAPER, SCISSORS game!
Enter your move: (r)ock (p)aper (s)cissors r
ROCK versus...
ROCK It is a tie!
Enter your move: (r)ock (p)aper (s)cissors r
ROCK versus...
PAPER You lose!
Enter your move: (r)ock (p)aper (s)cissors r
ROCK versus...
```

```
PAPER
You lose!
Enter your move: (r)ock (p)aper (s)cissors r
ROCK versus...
SCISSORS
You win!
You have 1 ties and 2 losses before your first win.
```

(Sample Run 2, **bold** is input from keyboard)

```
Welcome to ROCK, PAPER, SCISSORS game!
Enter your move: (r)ock (p)aper (s)cissors s
SCISSORS versus...
ROCK
You lose!
Enter your move: (r)ock (p)aper (s)cissors s
SCISSORS versus...
PAPER
You win!
You have 0 ties and 1 losses before your first win.
```

2. (50%) Name your Jupyter notebook `cos_approx.ipynb`. Using a while loop to implement an approximation of cosine function with polynomial:

$$\cos(x) \approx 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \frac{x^8}{8!} - \dots + \frac{x^{24}}{24!}$$

Compute the right-hand side for powers up to $N = 24$. (hint: You can import the function `factorial(N)` in `math` to return $N!$).

Below is a sample output:

```
cos(2.3) approximation is -0.6662760212798241
```

- Bonus (20%)** Name your file `RemoveOutliers.ipynb`. When analyzing data collected as part of a science experiment it may be desirable to remove the most extreme values before performing other calculations. Write a function `remove_outliers` that takes a list of values and a non-negative integer, n , as its parameters. The function should return a list with the n largest elements and the n smallest elements removed and a list of outliers. The order of the elements in the returned lists does not have to match the order of the elements in the original list.

Write a Python program that demonstrates your function. It should prompt the user to input the number n , a few data, signify the end of input with 'q' or 'Q', and remove the n largest and n

smallest values from it by calling the function described previously. Display the original list, the list with the outliers removed, and the outliers. Below is a sample run:

```
Enter the number of smallest and largest values to remove: 2
Enter a value (q or Q to quit): 1001
Enter a value (q or Q to quit): 2001
Enter a value (q or Q to quit): 1
Enter a value (q or Q to quit): 2
Enter a value (q or Q to quit): -1001
Enter a value (q or Q to quit): -20
Enter a value (q or Q to quit): 5
Enter a value (q or Q to quit): 4
Enter a value (q or Q to quit): 3
Enter a value (q or Q to quit): 1
Enter a value (q or Q to quit): 2
Enter a value (q or Q to quit): 3
Enter a value (q or Q to quit): q
The original data: [1001, 2001, 1, 2, -1001, -20, 5, 4, 3, 1, 2, 3]
The data with the outliers removed: [1, 1, 2, 2, 3, 3, 4, 5]
The outliers: [-1001, -20, 1001, 2001]
```