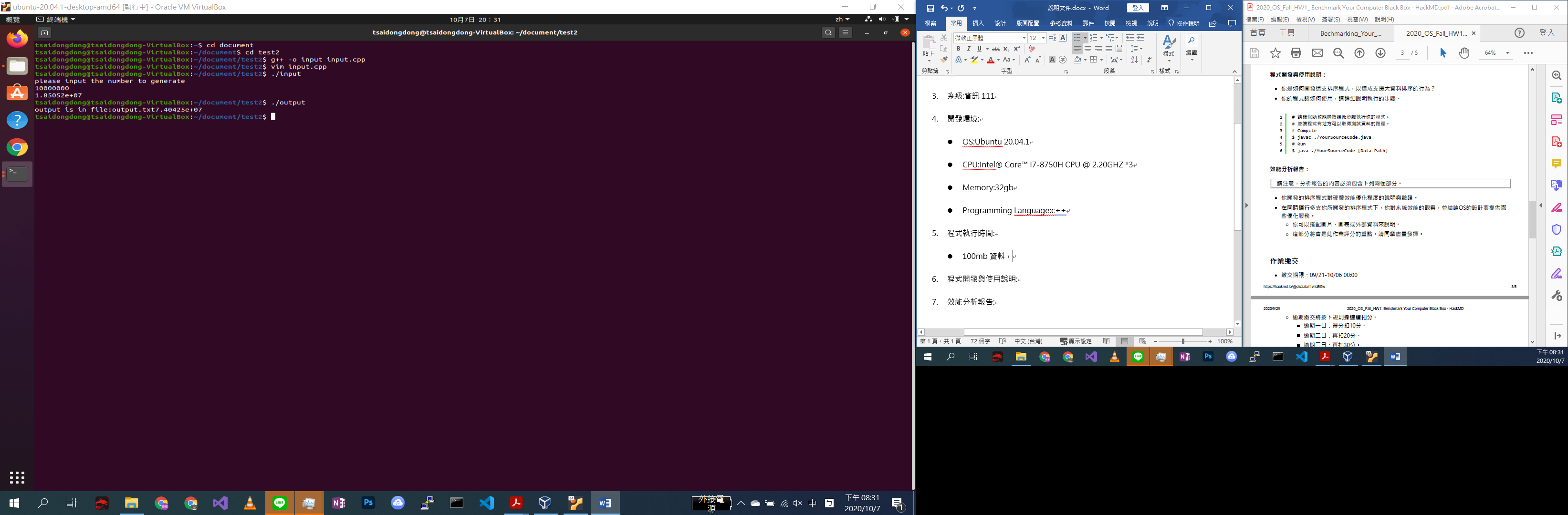
1. 姓名:蔡東霖
2. 系級:資訊111
3. 開發環境:

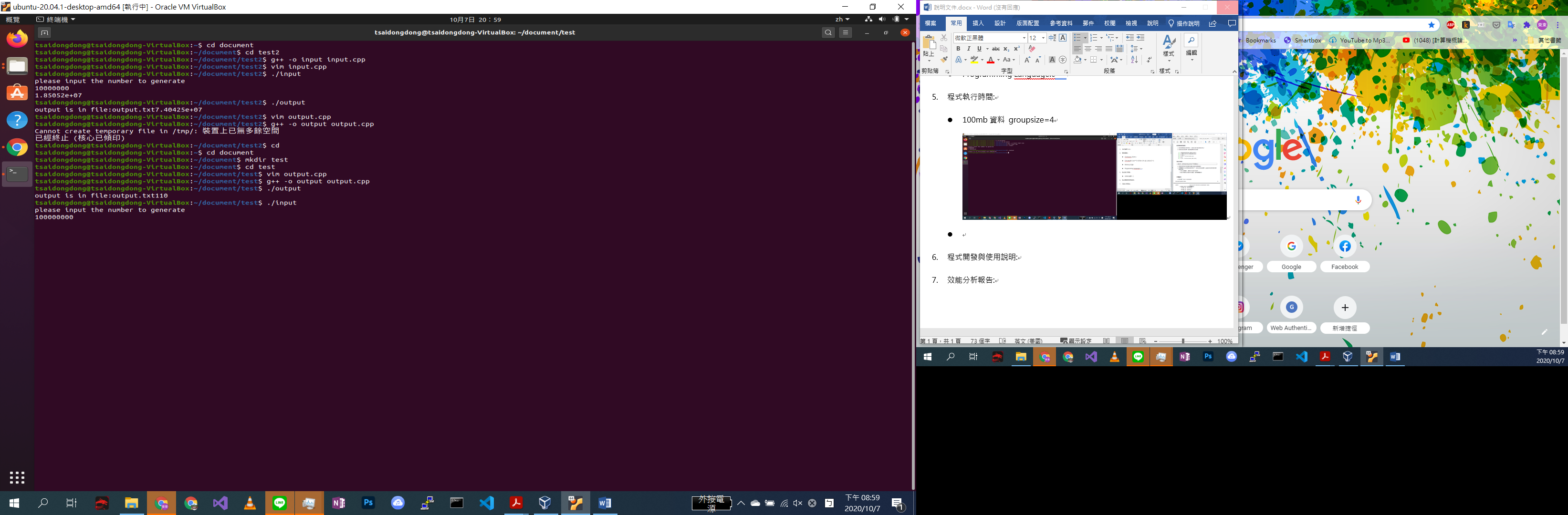
* OS:Ubuntu 20.04.1
* CPU:Intel® Core™ I7-8750H CPU @ 2.20GHZ \*3
* Memory:32gb
* Programming Language:c++

1. 程式執行時間:

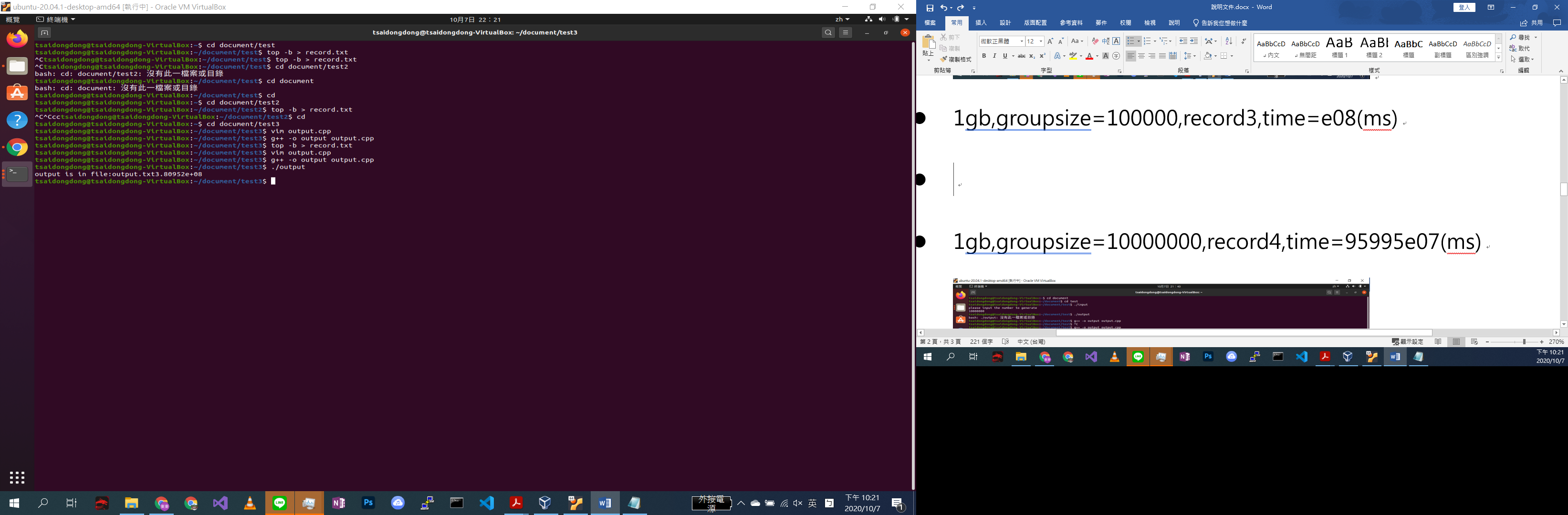
* 100mb資料 groupsize=4, ,time=1.85852e7(ms)



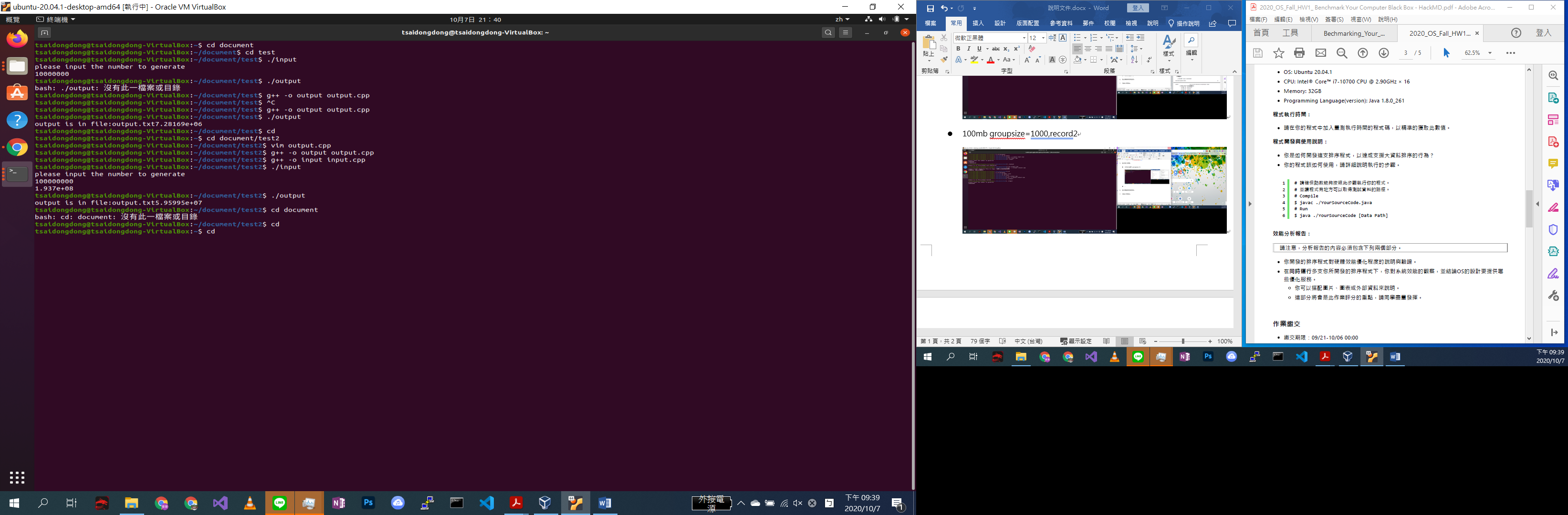
* 100mb groupsize=1000time=110(ms)



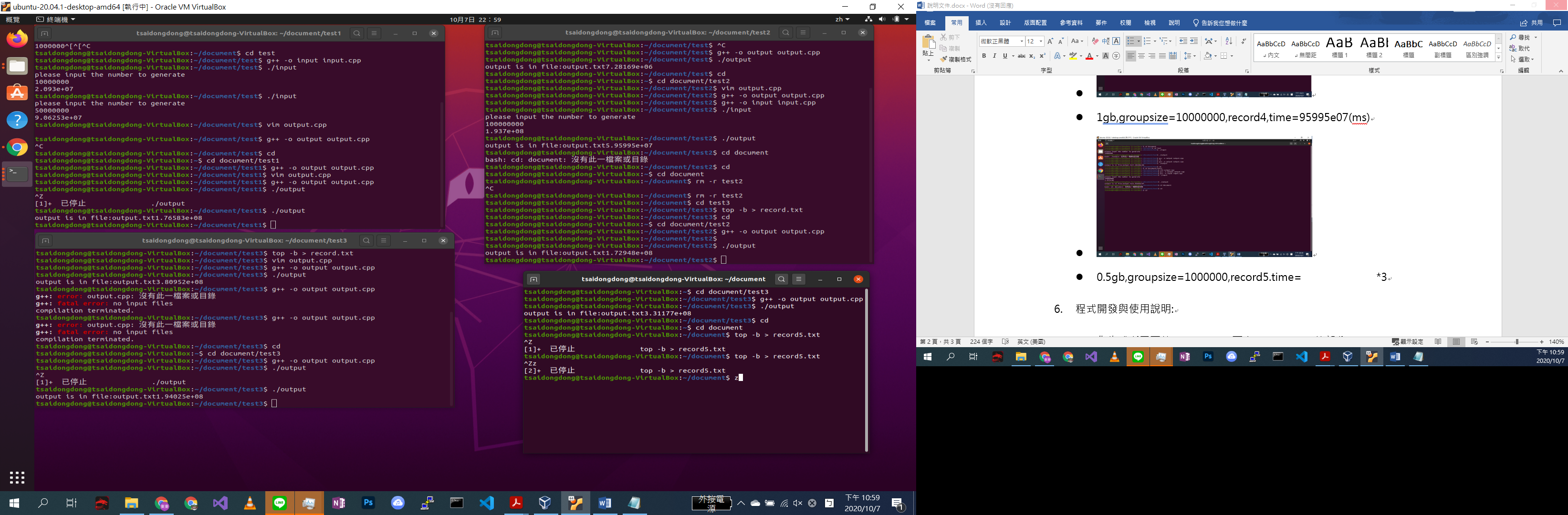
* 1gb,groupsize=100000,time=80952e08(ms)



* 1gb,groupsize=10000000,time=95995e07(ms)



* 0.5gb,groupsize=1000000 \*3

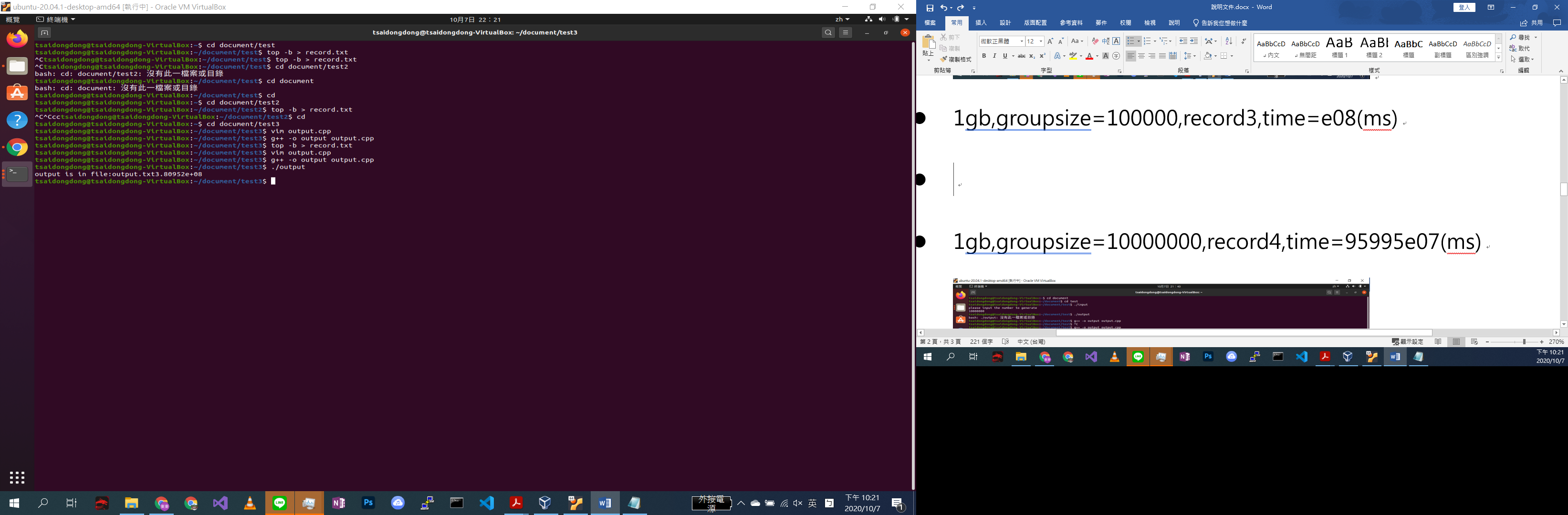


* 0.5gb,groupsize=100000000 \*3

(忘記截圖qq)

1. 程式開發與使用說明:

* 先生成所需要的input.cpp，再寫output.cpp的部分
* Output.cpp的部分，先將整個檔案分成幾個group，分類方式是指定一個group可以有幾個東西，最後是設定10000000，測試過後覺得最好的效率
* 先分每一個group內建的sort，再把每個goup產生的檔案使用minheap的方式分類
* 每次都先調出第一個值去排序，被發出來後group的下一個下去補，以類推直到所有的group都完



1. 效能分析報告:

* 使用單一程式在執行時，由上面的圖可以發現，單一程式執行時，因為給定的單一group的number太少時，電腦為了開檔閉檔而所花的時間是如此巨大的，但如果拉高單一一個group的時候，因為開檔閉檔的次數減少了，所花的時間就少很多，甚至快到無法想像。