1. **Your name and student ID**

   Name: 陳俊元 , Student ID: 110065536

2. **How to compile and execute your program and give an execution example.**

   in HW3/src:

   for generate the executable files "hw3" in "HW3/bin"

```
$make
```

   for remove the executable files

```
$make clean
```

   To run the files

```
../bin/<exe> <hardblock file> <net file> <pl file> <floorplan file> <deadspace ratio>
```

```
../bin/hw3 ../testcase/n100.hardblocks ../testcase/n100.nets ../testcase/n100.pl ../output/n100.floorplan 0.1
```

   To verify

```
../verifier/Verify <hardblock file> <net file> <pl file> <floorplan file> <deadspace ratio>
```

```
../verifier ../testcases/n100.hardblocks ../testcases/n100.nets ../testcases/n100.pl ../output/n100.floorplan 0.1
```

3. **The wirelength and the runtime of each testcase with the dead space ratios 0.15 and 0.1, respectively. Notice that the runtime contains I/O, constructing data structures, initial floorplanning, computing (perturbation) parts, etc. The more details your experiments have, the more clearly you will know where the runtime bottlenecks are. You can plot your results like the one shown below.**
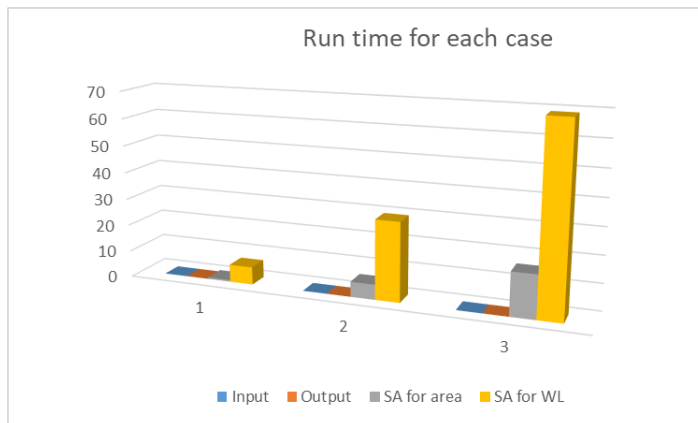
用了一個3D圖由於input output的時間佔比太小，較難用圖表展現
以下為每一個case的時間，以下dead space ratio皆為0.1

Testcase	Input(s)	Output(s)	SA for area(s)	SA for WL(s)
n100	0.0035	0.001	0.572	6.6
n200	0.0041	0.0014	5.65	29.43
n300	0.006	0.0013	16.075	68.95

4. **Please show that how small the dead space ratio could be for your program to produce a legal result in 10 minutes.**

Testcase	DSR	total WL	total time(s)
n100	0.09	292610	5.75
n200	0.08	565615	13.83
n300	0.095	615290	22.42

我發現最大的問題不是時間不夠，是因為我想要同時兼具wire length以及area，以上的表格為兩者兼具的情況，以上列出只考慮符合fixed outline的情況，也就是當初出現符合outline的結果及輸出答案，因此WL的成績會低很多。

透過這個實驗我發現最大的問題不是run time，問題比較像是是我的code有更好的優化空間，都還有充足的時間就結束，應該是少了很多soulution space的探索
詳細的實驗結果會放到此報告的附錄

5. **The details of your implementation. If there is anything different between your implementation and the algorithm in the DAC-86 paper, please reveal the difference(s) and explain the reasons.**

一開始實作演算法的時候遇到許多的問題，因此向助教詢問了一些意見，也實作了很多助教給予的想法，也發現不管在資料結構上，或者是執行時間上有非常大的優化

5-a. initialNPE的實作不同：講義上說一開始都使用12V3V….的初始化，但我發現這樣的擺放會讓程式多很多時間去先把hard blok擺放到一個及格的位子，因此我選擇先把block擺滿一個row再擺到下一個row，先讓他盡可能趨近於可以符合限制的樣子再去作調整，這樣的改變也讓程式的速度快非常非常多。

5-b. SA分成兩個階段，第一個階段是先擺area，第二個階段再去優化WL
其實邏輯跟5-a非常相似，都是希望先找到符合限制的解之後再去優化，否則同時有面積以及線長的變數要考慮，會讓程式執行的比較慢。

5-c. 在perturb上，講義上的M1為交換相鄰的block，然而我在實作的時候發現按照講義上的操作，我的SA都找不到正解，後來經由助教的建議，把M1改成global的hard

block皆可以交換才跑出正確的結果

5-d. 講義的t0有透過一個算式把他算出來, 我發現隨便給一個值最後出來的結果並不會相距太遠, 因此我一開始便直接把t0設為1000

5-e. 講義一開始把MT, uphill, reject皆設為0, 但我想寫進while loop中, 又因為while的中止條件為reject / MT 因此一開始MT不能為0, 我便給他1, 後來也發現程式並沒有甚麼問題

6. **What tricks did you do to speed up your program or to enhance your solution quality? Also plot the effects of those different settings like the ones shown below.**

以下我將以n200 DSR為0.1作為實驗數據:
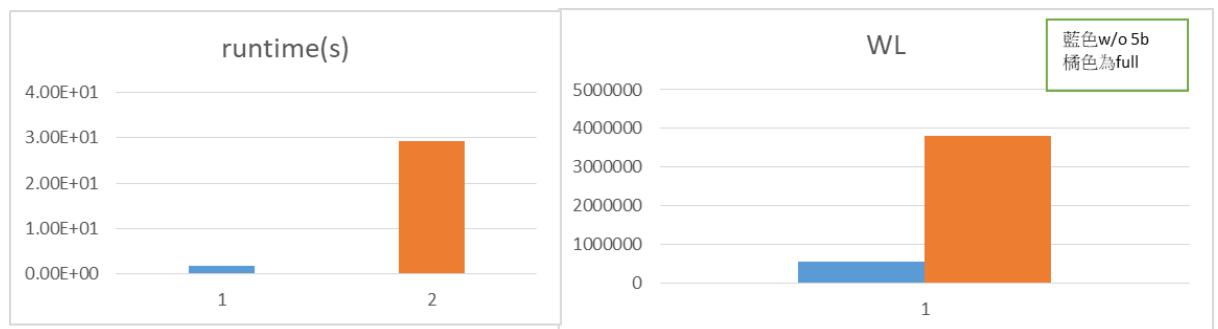
有趣的是我理論上應該也要實作w/o 5-a的圖表比較, 但不知道為何我使用講義的12V3V...SA便永遠跑不出結果, 我想也許是我的程式有更多需要改善的空間。

7. **Please compare your results with the previous top 3 students' results for the case where the dead space ratio is set to 0.15, and show your advantage either in runtime or in solution quality. Are your results better than theirs?**

   ✓ If so, please express your advantages to beat them.
   ✓ If not, it's fine. If your program is too slow, then what could be the bottleneck of your program? If your solution quality is inferior, what do you think that you could do to improve the result in the future? Top 3 students' results (dead space ratio = 0.15) Wirelength Runtime (s)
   我的成績與第三名的同學並沒有差距太大, 然而我的run time少了非常多, 我有試著調整各種參數, 但仍然沒有辦法擊敗前三名的同學, 我猜測可能是我的code探索的solution space不夠, 可以從我上述提到的一些觀點發現我的code應該是有一些bug但我沒有發現的, 改進的話應該可以超越前三位同學

Top 3 students' results (dead space ratio = 0.15)

Ranks	Wirelength			Runtime (s)		
	n100	n200	n300	n100	n200	n300
1	**207309**	**367785**	**504903**	**13.97**	**84.54**	263.33
2	209351	379674	521749	25.57	99.49	**209.78**
3	210220	392175	544879	37.45	105.83	486.73

```
[cychen21@ic51 HW3_grading]$ bash HW3_grading.sh
|------------------------------------------|
|                                          |
|     This script is used for PDA HW3 grading.  |
|                                          |
|------------------------------------------|
grading on 110065536:
  testcase  |      ratio  |  wirelength  |     runtime  |  status
      n100  |      0.15   |      218050  |       1.62   |  success
      n200  |      0.15   |      394470  |       8.03   |  success
      n300  |      0.15   |      566448  |      15.63   |  success
      n100  |      0.1    |      229848  |       1.85   |  success
      n200  |      0.1    |      403004  |      11.12   |  success
      n300  |      0.1    |      579595  |      29.88   |  success
```

8. **If you implement parallelization (for algorithm itself), please describe the implementation details and provide some experimental results**

我沒有實作平行化

9. **What have you learned from this homework? What problem(s) have you encountered in this homework?**

雖然此節課並不是教授C++的課程，但我在實作partition以及floorplan的時候學會到除了演算法之外還有C++非常多的技巧幫助我的程式跑的更少memory 以及更快，以下列出我此次作業學會的部分，許多地方其實都是大學時候應該要學好的部分。也很開心有這個機會可以重新練習自己的coding

a. make -o3 可以大幅提升program的執行速度

b. 應該要利用struct/class 把所要用到的結構都存起來，而不是像partrition一樣狂開一堆unorder_map

c. 除了檢查run time之外，由於此次作業有地方被工作站kill掉，回頭檢查才發現有個地方一直不斷的new node，但其實可以不需要，也寫了一個memory usage的程式發現自己的程式從原本需要40G以上的RAM因此被卡掉大幅降低到20M左右

這次的作業若是沒有同學以及助教的幫助我應該是無法完成，非常謝謝助教對於作業的很多提醒，不管是在coding上或者是對此演算法本身的探討

附錄：4-1
實驗目的：看看最低dead space ratio可以到多少

```
+ ../verifier/verifier ../testcases/n100.hardblocks ../testcases/n100.nets ../testcases/n100.pl ../output/n100.floorplan 0.09
Total block area: 179501
Width/Height of the floorplan region: 442
Wirelength: 292610
Checking fixed-outline and non-overlapping constraints of the blocks locating ...
WL computed by verifier: 292610 <---> WL reported in .floorplan: 292610
OK!! Your output file satisfies our basic requirements.
```
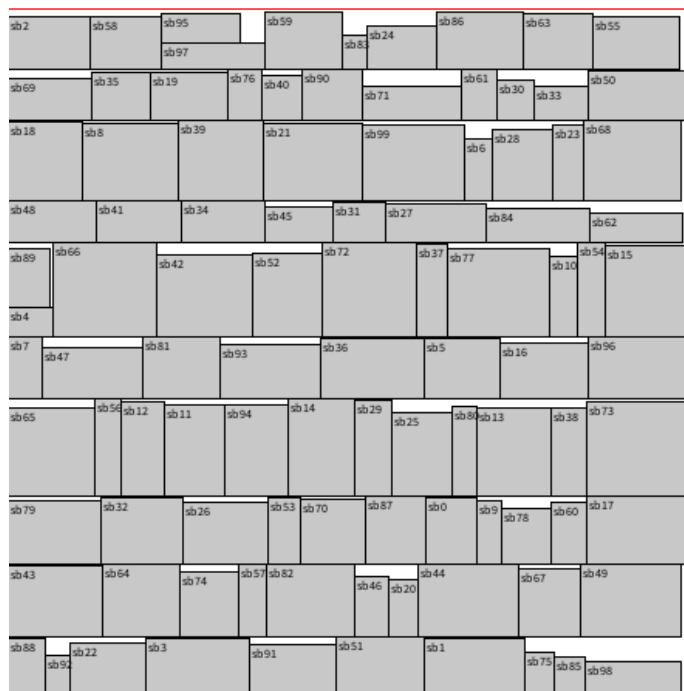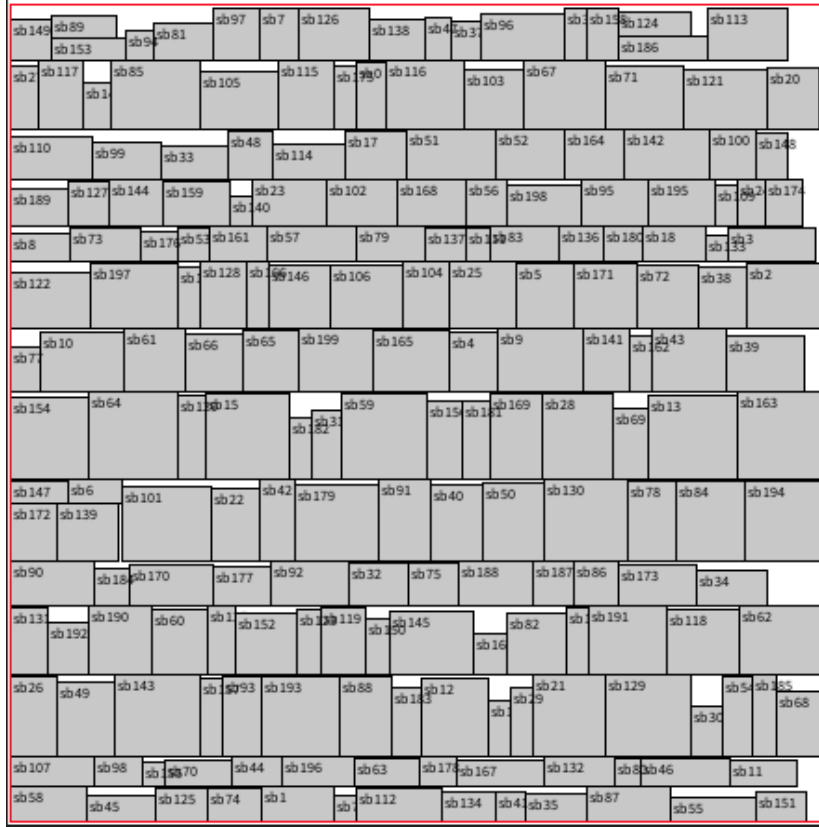
圖(1) n100於DSR 0.09的實驗結果

```
+ ../verifier/verifier ../testcases/n200.hardblocks ../testcases/n200.nets ../testcases/n200.pl ../output/n200.floorplan 0.08
Total block area: 175696
Width/Height of the floorplan region: 435
Wirelength: 565615
Checking fixed-outline and non-overlapping constraints of the blocks locating ...
WL computed by verifier: 565615 <---> WL reported in .floorplan: 565615
OK!! Your output file satisfies our basic requirements.
```

圖(2) n200於DSR 0.08的實驗結果

```
+ ../verifier/verifier ../testcases/n300.hardblocks ../testcases/n300.nets ../testcases/n300.pl ../output/n300.floorplan 0.095
Total block area: 273170
Width/Height of the floorplan region: 546
Wirelength: 615290
Checking fixed-outline and non-overlapping constraints of the blocks locating ...
WL computed by verifier: 615290 <---> WL reported in .floorplan: 615290
OK!! Your output file satisfies our basic requirements.
```

圖(2) n300於DSR 0.0095的實驗結果

以下為使用printer的結果
n100 DSR:0.1

n200 DSR:0.1

n300 DSR:0.1