**學號：R05943092**

**姓名：曾育為**

**實作 (Implementation)**

* 1. **演算法流程 (Algorithm Flow)**

演算法流程圖

配合流程圖，將程式運作流程再加上文字敘述如下：

1. 讀入檔案，將cell和net各自建立資料結構，並將net連結所觸及的cell，net連結所有觸及的net
2. 取最大cell ID切半，分成左右二堆，並依據左右二邊的cell分佈及net連線，初始化cell的gain及cut數量
3. 建立一個bucket list，並將cell的gain初始化的結果填入bucket list中
4. 取最大gain並檢查是否維持balance function，如符合平衡，則將cell移到另外一邊，反覆進行cell總數的次數則停止
5. 在cell總數次數中，取出最好的結果，並輸出結果
   1. **虛擬碼 (Pseudocode)**

**Input:** 、

**Output:** Cutsize、G1、G2

**begin**

Divide all cell into two group. Initialize all the gain value of cell.

Build bucket list to record all the gain value of cell.

**for**( each cell )

update all the gain value of cell.

Check the balance after the cell move

Get the optimal gain value

Output the cut size and left number、right number

**end**

* 1. **資料結構 (Data Structure)**

下面將介紹程式中主要的資料結構：

1. Cell: 代表輸入檔的cell

class Cell

{

public:

unsigned \_name;//紀錄cell本身的ID

string \_side;//紀錄cell在左邊or右邊

int \_gain;//紀錄cell的gain值

bool \_locked;//紀錄cell是否固定

vector<unsigned> \_net\_id;//紀錄cell連接net的id

};

1. Net: 代表輸入檔的net

class Net

{

public:

unsigned \_name;// 紀錄net本身的ID

unsigned \_l\_num;//紀錄左邊區域此net所連結的cell個數

unsigned \_r\_num;//紀錄右邊區域此net所連結的cell個數

vector<unsigned> \_cell\_id; //紀錄net連接cell的id

};

* 1. 實行結果：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Input\_0 | Input\_1 | Input\_2 | Input\_3 | Input\_4 | Input\_5 |
| Balance factor | 0.2 | 0.01 | 0.02 | 0.1 | 0.01 | 0.01 |
| Net number | 166998 | 5000 | 10000 | 88888 | 166998 | 483599 |
| Cell number | 150750 | 3000 | 7000 | 66666 | 150750 | 382489 |
| Cut number | 14871 | 1492 | 2642 | 31855 | 54614 | 170000 |
| Runtime(S) | 103.612 | 0.087 | 0.3748 | 62.0489 | 318.418 | 3143.55 |