**Physical Design PA4-Routing**

**GIEE**

**學號：R05943092**

**姓名：曾育為**

* 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Pin建立資料結構。
2. 將Pin依據x座標進行排列，將每個pin連結sorting後此pin往後20個pin。
3. 利用Kruskal Algorithm找出Minimum spanning tree

(a). 根據曼哈頓距離，由小到大進行排序

(b). 將每條net先設定成一個cluster，不斷地結合，最後得到一個cluster，完成minimum spanning tree

(iv). 將minimum spanning tree所有的net取rectilinear並分類成V-line和H-line

(vi). 輸出結果

* 1. **資料結構 (Data Structure)**

下面將介紹程式中主要的資料結構：

1. Pin: 代表輸入檔的Pin資訊

class Pin

{

string \_name;//Pin的名稱

int \_id;//Pin的ID

int \_x;//Pin的x座標

int \_y;//Pin的y座標

int cluster\_num;//在進行kruskal algorithm的cluster ID

};

1. Net: 代表根據二個Pin相連所得的net

class Net

{

Pin \*pin\_l;//連接一個pin

Pin \*pin\_r;//連接另一個pin

int \_m\_distance;//二個pin之間的曼哈頓距離

};

1. F\_net: 代表最後輸出檔的net

class F\_net

{

int pin\_l\_x;//pin\_l的x座標

int pin\_l\_y;//pin\_l的y座標

int pin\_r\_x;//pin\_r的x座標

int pin\_r\_y;//pin\_r的y座標

bool v\_line;//判斷是否為V\_line或是H\_line

};

（iv）Cluster : Kruskal algorithm判別的依據

class Cluster

{

int \_id;//cluster的ID

vector<Pin\*> \_pin\_set;//儲存此cluster所有的pin

};

（vi）Router : 代表此Router的資訊

class Router

{

private :

fstream fp;//讀檔與輸出檔

int \_*boundary*\_x;//boundary的x

int \_*boundary*\_y;//boundary的y

int \_*number*\_pin;//pin的總個數

int total\_distance;//

vector<Pin> \_pin;//儲存所有的pin

vector<Net> \_net;//儲存所有的net

vector<F\_net> Final;//儲存所有的F\_net

};

* 1. **問題與討論：**

**(3-1). 輸出結果：**

* **固定參數:**
  + **取此pin點後20個pin連結成spanning tree**
* **實驗數據：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Case** | **Pin number** | **Final net** | **Wire length** | **Time** |
| **case1** | **10** | **17** | **276** | **0.001019** |
| **case2** | **50** | **91** | **539** | **0.00185** |
| **case3** | **100** | **176** | **814** | **0.002973** |
| **case4** | **500** | **971** | **18318** | **0.01051** |
| **case5** | **1000** | **1997** | **2891959** | **0.018083** |
| **case6** | **2000** | **3998** | **47861840** | **0.035662** |
| **case7** | **5000** | **9998** | **1048543631** | **0.089395** |
| **case8** | **10000** | **19998** | **19982245272** | **0.208342** |

**(3-2). Spanning graph建立net數量探討：**

* **固定參數：**
  + **以case8作為比較基準**
  + **Pin number固定**
* **實驗數據：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **取此pin點後幾個pin點** | **Spanning Graph’s net number** | **Wire length** | **Final net** | **Time** |
| **10** | **189955** | **407,2578,2302** | **19998** | **0.133851** |
| **20** | **1189810** | **199,8224,5272** | **19998** | **0.208342** |
| **30** | **1289565** | **140,1540,7498** | **19998** | **0.258761** |
| **40** | **1389220** | **113,0896,2233** | **19998** | **0.315857** |
| **50** | **1488775** | **98,3902,1548** | **19998** | **0.377045** |
| **100** | **1985050** | **79,0995,3118** | **19998** | **0.667634** |
| **1000** | **19490500** | **77,4873,2129** | **19998** | **6.08939** |

* **結論：**

**Spanning Graph如果能取越多條net所得的Wire length結果越短，相對地所耗費的時間也會越久，但如果取超過100個所改善的幅度將逐漸縮小。**