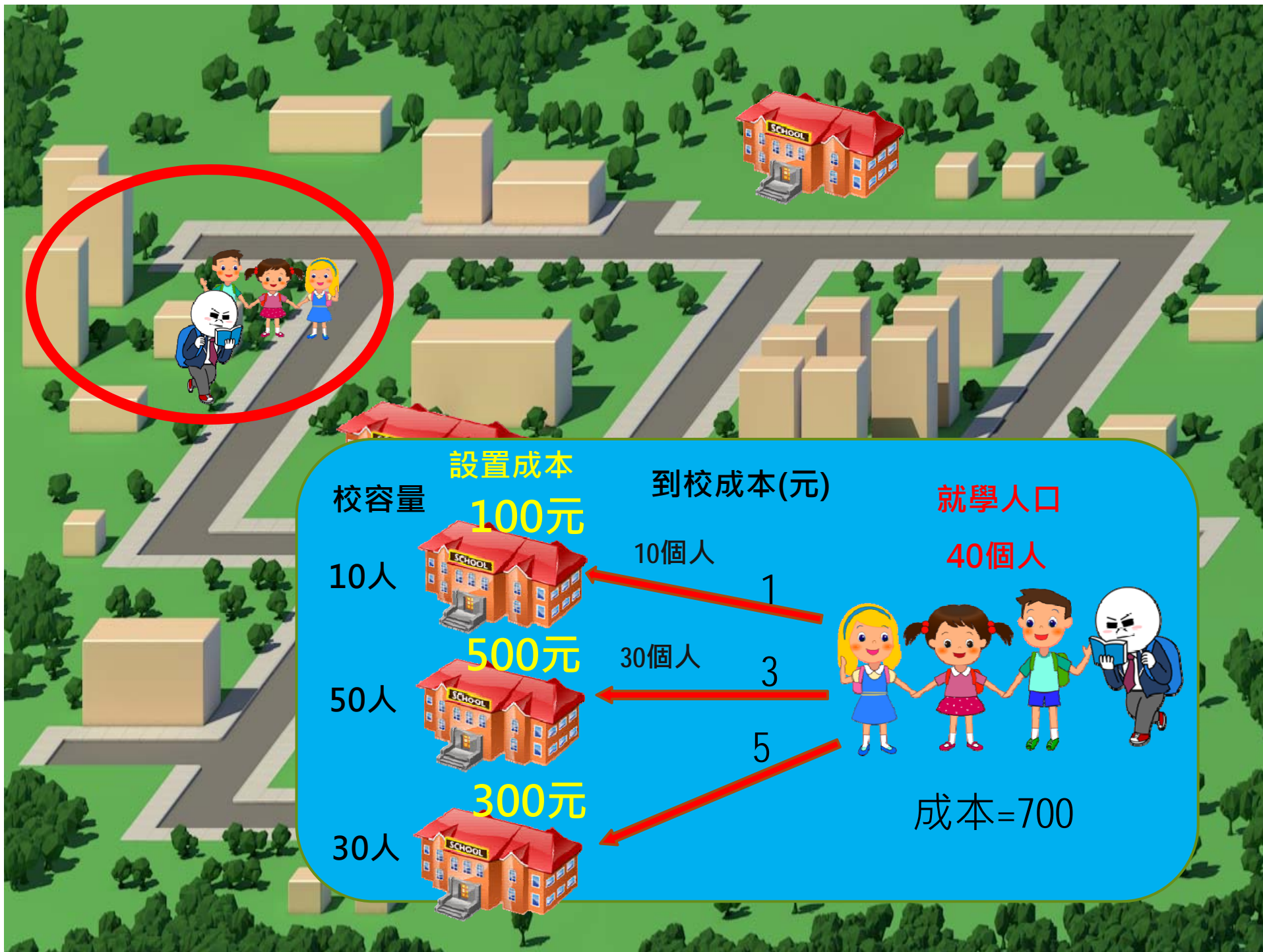
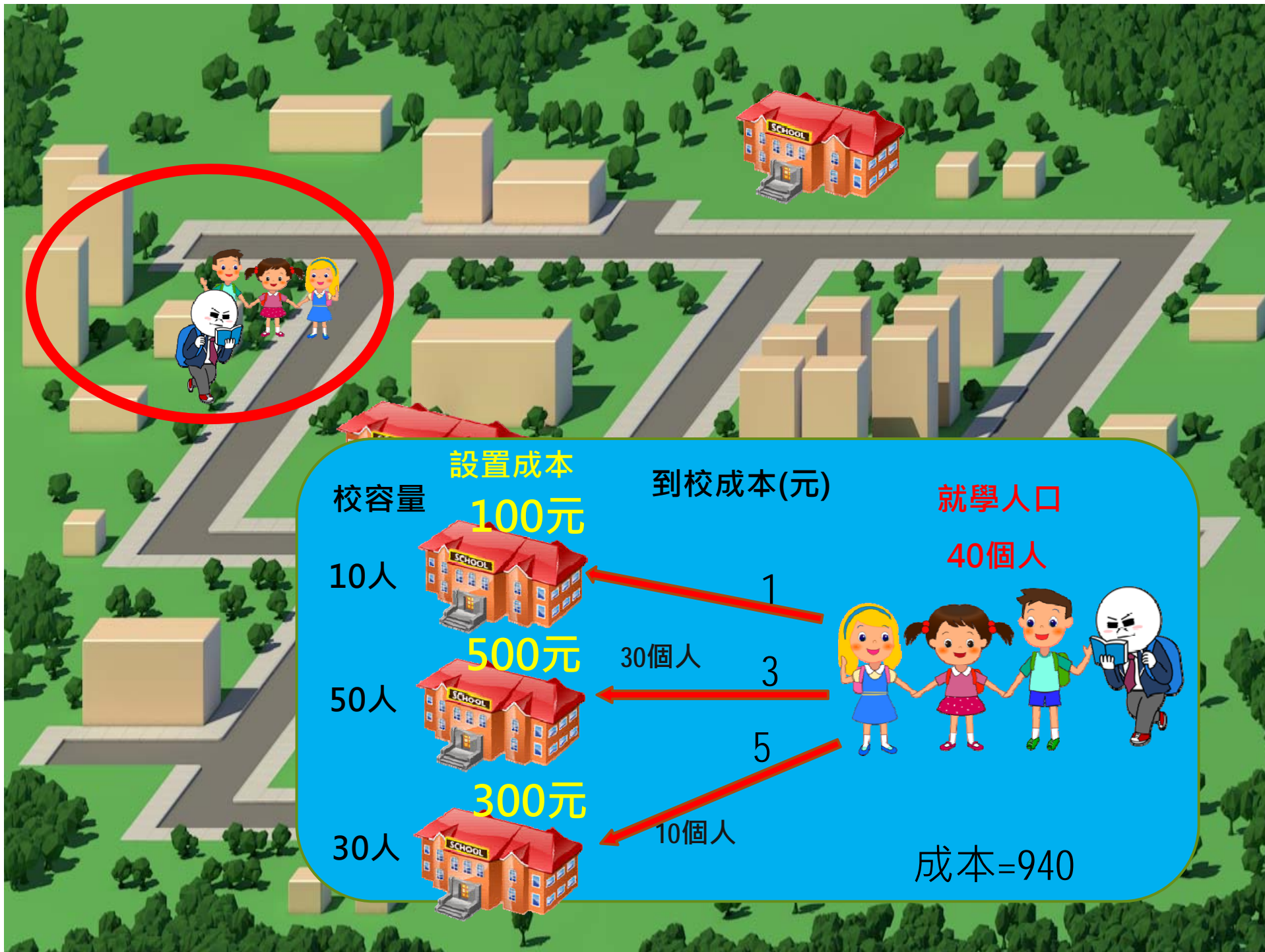


The background features abstract green geometric shapes, including triangles and polygons, some with a fine grid pattern, set against a white background.

設施定位問題問題 (Facility Location Problem, FLP)







兩個決策：

1. 設施開啟或關閉
2. 決定開啟設施後如何運輸



FLP with alternative paths

FLP with alternative paths

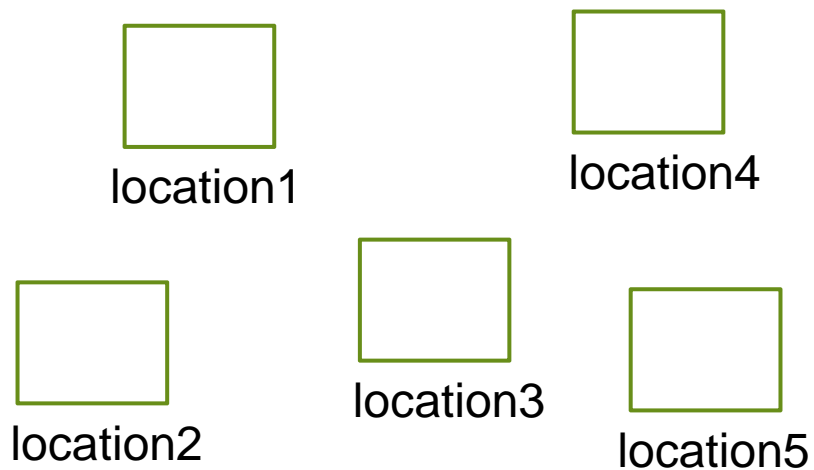
- 5 machines



- 4 products



- 5 locations



Distance matrix between locations

$$= \begin{bmatrix} 0 & 6 & 17 & 4 & 10 \\ 6 & 0 & 8 & 5 & 11 \\ 17 & 8 & 0 & 14 & 2 \\ 4 & 5 & 14 & 0 & 13 \\ 10 & 11 & 2 & 13 & 3 \end{bmatrix}$$

FLP with alternative paths



location1



location4



location2



location3



location5

FLP with alternative paths



location1



location4



location2



location3



location5

How to locate?

FLP with alternative paths



- 產品1

- 產量1,500
- 生產可替代路徑
 1. 1,2,3
 2. 1,2,4
 3. 1,2,5
(機台1 → 機台2 → 機台5)



- 產品3

- 產量200
- 生產可替代路徑
 1. 1,2,3,4
 2. 1,3,4,5



- 產品2

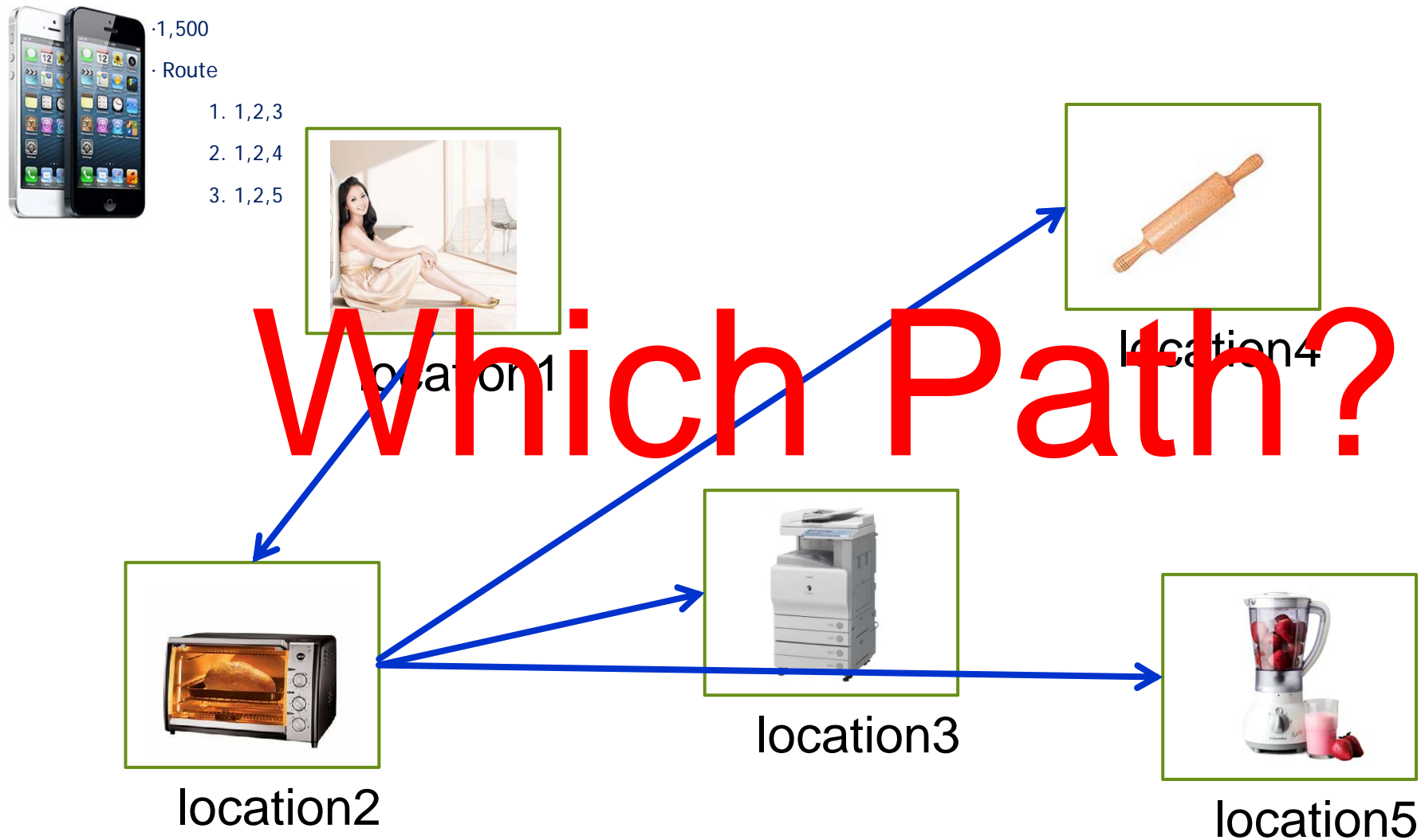
- 產量5,000
- 生產可替代路徑
 1. 1,2,3,4
 2. 2,3,4,5



- 產品4

- 產量40,000
- 生產可替代路徑
 1. 1,2,3,4,5

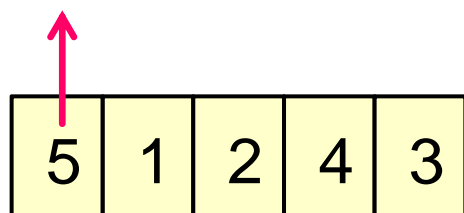
FLP with alternative paths



解的編碼

- 安排5個機台在5個機台可置放位置，
4個產品要靠這些機台生產

「機台5」放在「位置1」



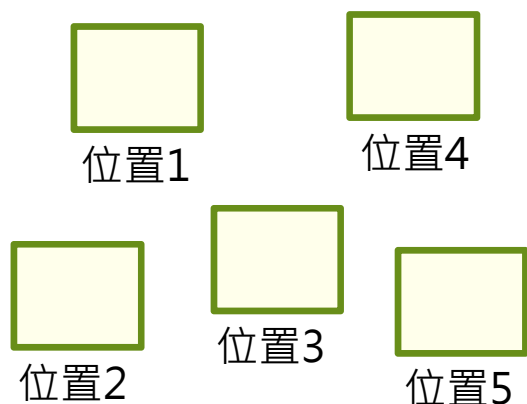
Part 1 (5 個位置)
(1~5的隨機排列)

「產品1」選「Route 2」來生產
(機台1 → 2 → 4)



Part 2 (4個產品)
(各產品給予一個不超過
路徑個數上界的隨機數)

Product	Processing route		
	1	2	3
1	1,2,3	1,2,4	1,2,5
2	1,2,3,4	2,3,4,5	-
3	1,2,3,4	1,3,4,5	-
4	1,2,3,4,5	-	-

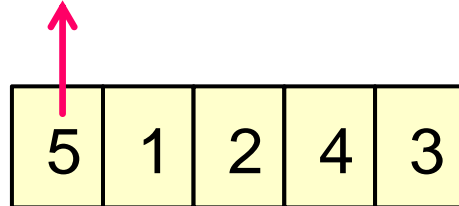


Python code for solution representation

- 安排5個機台在5個機台可置放位置
4個產品要靠這些機台生產

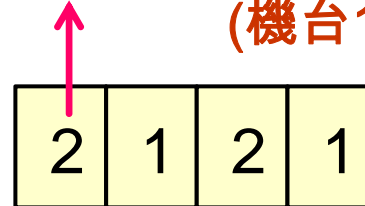
Product	Processing route		
	1	2	3
1	1,2,3	1,2,4	1,2,5
2	1,2,3,4	2,3,4,5	-
3	1,2,3,4	1,3,4,5	-
4	1,2,3,4,5	-	-

「機台5」放在「位置1」



Part 1 (5 個位置)

「產品1」選「Route 2」來生產
(機台1 → 2 → 4)



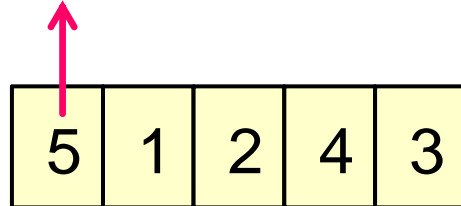
Part 2 (4個產品)

```
45 def initPop():                                # 初始化群體
46     p1 = []
47     p2 = []
48
49     for i in range(NUM_CHROME) :
50         # ==== Step 2-1. 產生0~4的隨機排列 ====
51         p1.append(np.random.permutation(NUM_MACHINE))
52
53         # ==== Step 2-2. 產生 NUM_PRODUCT 個上下界的路徑選擇 ====
54         p2.append([np.random.randint(len(route[j])) for j in range(NUM_PRODUCT)])
55
56     return p1, p2
```

解的解碼

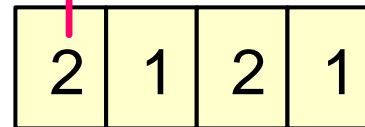
- 安排5個機台在5個機台可置放位置，
4個產品要靠這些機台生產

「機台5」放在「位置1」



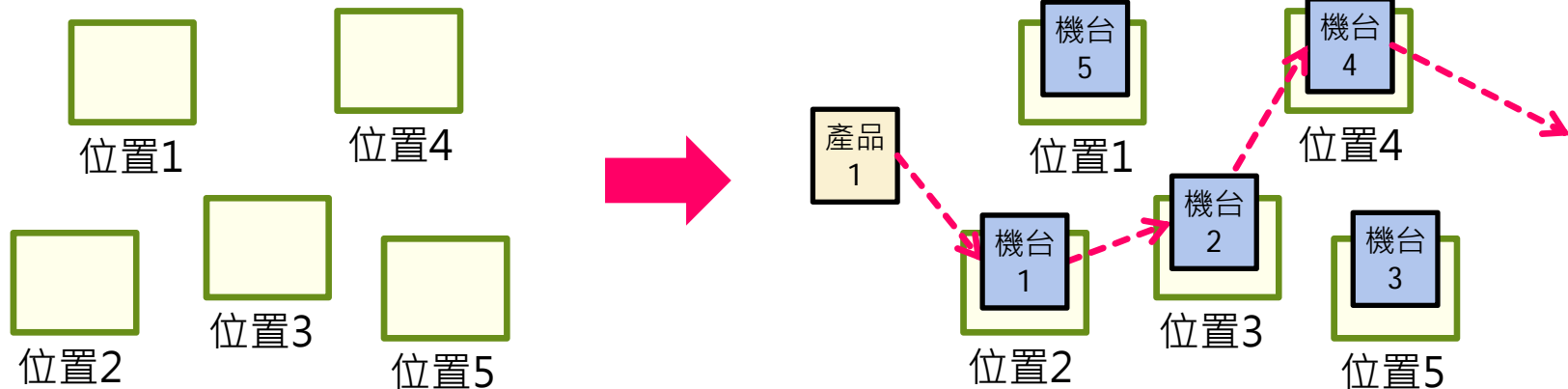
Part 1 (5 個位置)

「產品1」選「Route 2」來生產
(機台1 → 2 → 4)



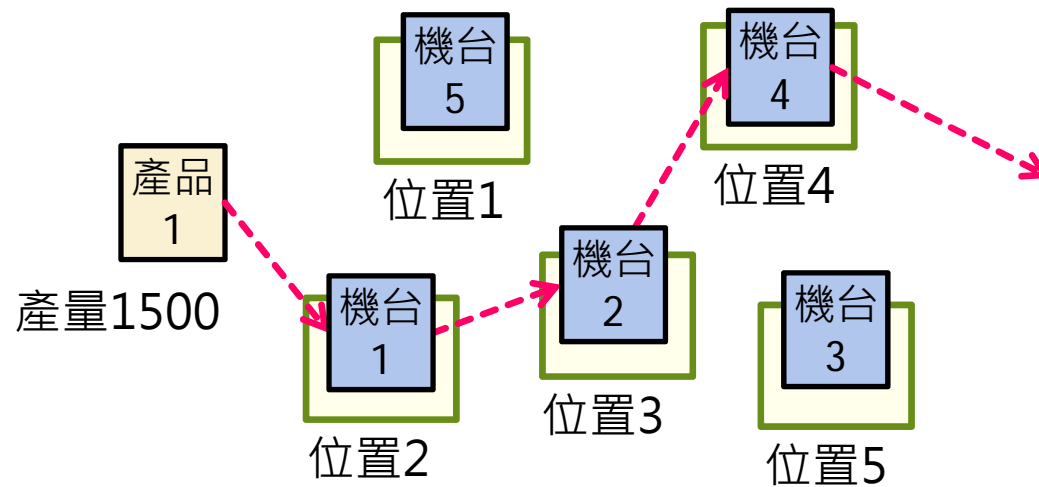
Part 2 (4個產品)

Product	Processing route		
	1	2	3
1	1,2,3	1,2,4	1,2,5
2	1,2,3,4	2,3,4,5	-
3	1,2,3,4	1,3,4,5	-
4	1,2,3,4,5	-	-



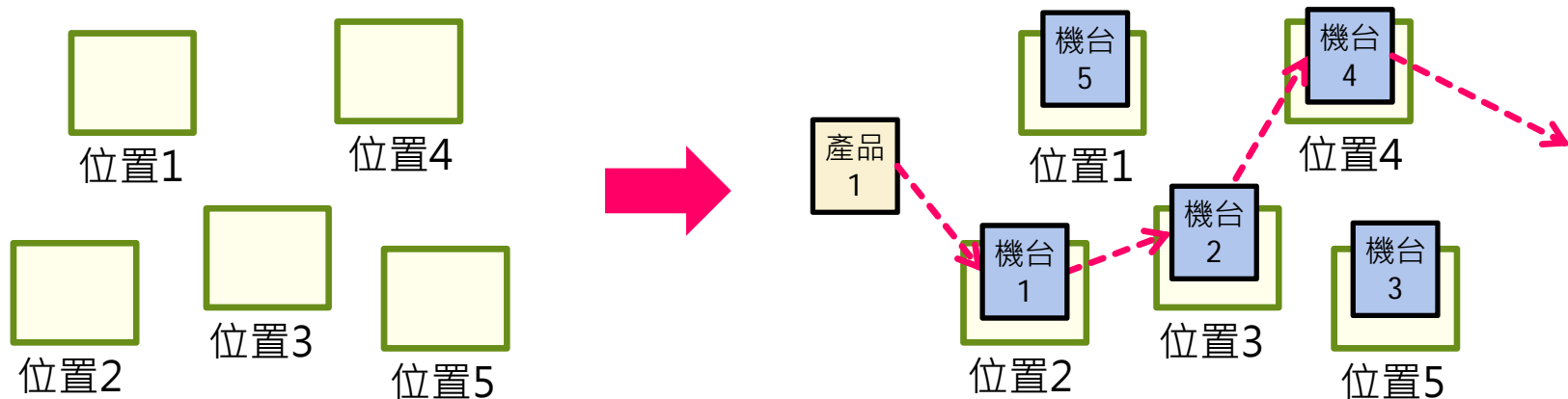
成本函數

- Cost = 產品1的產量 x 產品1走的路徑
+ 產品2的產量 x 產品2走的路徑
+ 產品3的產量 x 產品3走的路徑
+ 產品4的產量 x 產品4走的路徑



Python code for fitness evaluation

```
45 def fitFunc(x1, x2):          # 適應度函數      # ==== Step 4. 改適應度函數 ====
46     totalCost = 0
47
48     for i in range(NUM_PRODUCT):
49         myRoute = route[i][x2[i]]
50         pre_j = myRoute[0]
51         tmpDist = 0
52         for j in range(1, len(myRoute)):
53             tmpDist += dist[x1[pre_j]][x1[myRoute[j]]]
54             pre_j = myRoute[j]
55         totalCost += tmpDist * productVol[i]
56
57     return -totalCost          # 因為是最小化問題
```



Exercise

- 「GA11-facilityLocation-ex.py」是上述問題(5個位置、4個產品)的程式
- 請用此程式來求解下列問題
 - 15個位置、8個產品
 - 8個產品的產量300, 400, 200, 500, 100, 150, 250, 450
 - 8個產品各自的生產可行路徑(注意到程式中索引從0開始)

Product	Processing route		
	1	2	3
1	1,2,3,4,5,6	1,2,3,4,5,8,9,10	-
2	1,2,3,5,7,8,10,11	3,4,5,6,7,8,9,10	-
3	1,4,10,11	1,4,10,12,13,14	1,4,9,12,13,14
4	1,2,3,4,5,6,7,8,9,10,11,12,13,14	-	-
5	4,5,6,7,8,14,15	4,5,6,7,12,13,14,15	4,5,7,9,10,11,12,13,14,15
6	2,3,4,5,6	2,3,4,5,7,8	2,3,4,5,7,10
7	4,5,7,8,10,11,12	4,5,8,9,10,12,14	4,5,8,10,11,12,13,15
8	1,2,3,4,5,6,7,8,9,10,12	1,2,3,4,5,6,7,8,9,10,11,13,14	-

Exercise (cont.)

- 15個位置的距離矩陣

0	1	11	4	9	2	8	5	12	7	8	3	10	4	8
1	0	6	1	9	10	3	11	6	8	3	4	12	5	9
11	6	0	2	7	12	5	9	4	3	7	6	11	7	1
4	1	2	0	11	3	12	6	9	2	8	6	3	13	4
9	9	7	11	0	2	7	3	4	10	4	9	8	1	7
2	10	12	3	2	0	10	12	55	3	7	3	9	10	6
8	3	5	12	7	10	0	1	9	13	5	7	3	9	12
5	11	9	6	3	12	1	0	5	8	1	10	4	12	6
12	6	4	9	4	55	9	5	0	9	3	8	7	12	5
7	8	3	2	10	3	13	8	9	0	4	9	12	3	7
8	3	7	8	4	7	5	1	3	4	0	7	11	2	9
3	4	6	6	9	3	7	10	8	9	7	0	4	8	3
10	12	11	3	8	9	3	4	7	12	11	4	0	1	9
4	5	7	13	1	10	9	12	12	3	2	8	1	0	12
8	9	1	4	7	6	12	6	5	7	9	3	9	12	0

- 表格和矩陣均在「GA12-FLP-ex-inst.docx」檔案
- 請調參數以找出最佳解是33800

Appendix

「GA11-facilityLocation-ex.py」的演算法
的完整步驟

完整的步驟

● Step 1. 設定參數

- 4個產品、5台機器
- 4個產品的產量(1500, 5000, 200, 40000)
- 4個產品的各替代路徑

Product	Processing route		
	1	2	3
1	1,2,3	1,2,4	1,2,5
2	1,2,3,4	2,3,4,5	–
3	1,2,3,4	1,3,4,5	–
4	1,2,3,4,5	–	–

- 5個機器可擺放位置的 5 x 5 距離矩陣

0	6	17	4	10
6	0	8	5	11
17	8	0	14	2
4	5	14	0	13
10	11	2	13	3

完整的步驟

- Step 2. 設定初始解

- Part 1是5個可行位置 (隨機給0~4的排列)
- Part 2是 4 個產品的可行路徑ID (隨機給可行上下界)
 - ✓ 產品1有3種可行路徑、產品2有2種、產品3有2種、產品4有1種

- Step 3. 改mutation的任選part 2的基因的上下界

- 產品1有3種可行路徑、產品2有2種、產品3有2種、產品4有1種

- Step 4. 改適應度函數

- 根據 $x1[0], x1[1], \dots, x1[4]$ 來決定機台位置，然後根據這些位置和 $x2$ 的路徑選擇來計算繞行路徑
- 路徑乘上經過的產品數量即為總成本
- 回傳負的總成本，方可最小化總成本