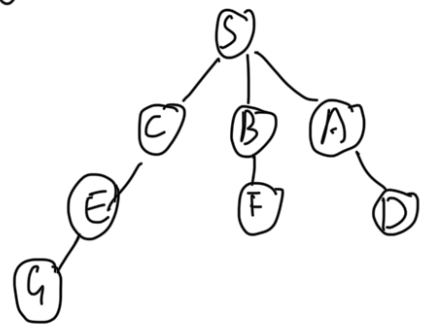
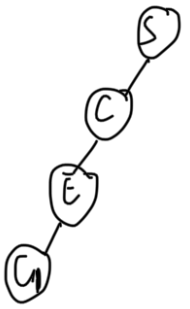
一 1.1 The binary search tree for BFS is



visited points : S, C, B, A, E, F, D, G

solution cost : 9 + 12 + 28 = 49

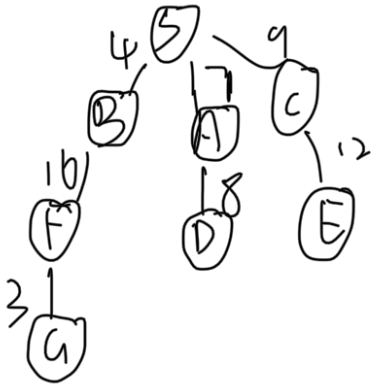
1.2 The binary search tree for DFS is



visited points : S, C, E, G

solution cost : 9 + 12 + 28 = 49

1.3 The binary search tree for UCS is



visited points : S, B, A, C, D, F, E, G

solution cost : 4 + 16 + 3 = 23

二 2.1 Represent solutions by positive intergers.

For example, O, A, B, C, D, E is represented as 1, 2, 3, 4, 5, 6. The solution O → A → C → E → D → B

will be represented as 1 2 4 6 5 3

2.2 The fitness function could be the minus tour length. Shorter tour length will lead to higher fitness

2.3 crossover operator: partically-mapped crossover:

① This operator first randomly select two cut points on both parents

② In order to creat an offspring, the substring between the two cut points in the first parent replace the corresponding substring in the second parent.

③ Then, the inverse replacement is applied outside of the cut points. In order to eliminate duplicates and recover all cities.

Example: parent 1: 1 |2 4|6 5 3

parent 2: 1|3 4|6 5 2

offspring:

step 1 : 1 2 4 6 5 2

step 2 : 1 2 4 6 5 3

A valid offspring 1 2 4 6 5 3 is generated by crossover operator

2.4 mutation operator: inversion based mutation

Randomly select two points on the tour, revers the substring between the selected points.

Example: tour: 1 2|4 6 5|3

After mutation: 1 2 5 6 4 3

A valid tour 1 2 5 6 4 3 is generated by mutation operator

三 Two parent tours 1 3 6 5 4 2 8 7 and 1 4 2 3 6 5 7 8

city 1 has edges to: 3, 4, 7, 8

city 2 has edges to: 3, 4, 8

city 3 has edges to: 1, 2, 6

city 4 has edges to: 1, 2, 5

city 5 has edges to: 4, 6, 7

city 6 has edges to: 3, 5

city 7 has edges to: 1, 5, 8

city 8 has edges to: 1, 2, 7

(a) City 1 is selected

city 2 has edges to: 3, 4, 8

city 3 has edges to: 2, 6

city 4 has edges to: 2, 5

city 5 has edges to: 4, 6, 7

city 6 has edges to: 3, 5

city 7 has edges to: 5, 8

city 8 has edges to: 2, 7

(b) City 3 is selected

city 2 has edges to: 4, 8

city 4 has edges to: 2, 5

city 5 has edges to: 4, 6, 7

city 6 has edges to: 5

city 7 has edges to: 5, 8

city 8 has edges to: 2, 7

(c) City 6 is selected

city 2 has edges to: 4, 8

city 4 has edges to: 2, 5

city 5 has edges to: 4, 7

city 7 has edges to: 5, 8

city 8 has edges to: 2, 7

(d) City 2 is selected

city 4 has edges to: 5

city 5 has edges to: 4, 7

city 7 has edges to: 5, 8

city 8 has edges to: 7

(e) City 4 is selected

City 5 has edges to: 7

City 7 has edges to: 5, 8

City 8 has edges to: 7

(f) City 5 is selected

City 7 has edges to: 8

city 8 has edges to: 7

(g) City 7 is selected

city 8 has edges to:

The final tour is 1 3 6 2 4 5 7 8