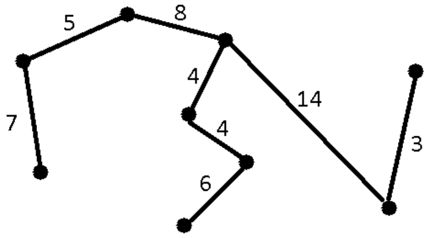
97 交通大學　資結與演算法

1. F T T F F
2.
   (1) 若 G is a connected weighted graph, 可產生大於等於一個 spanning tree, 其中 weight 和為最小者即為 minimum spanning tree
   (2) 挑邊順序為(weight)：3, 4, 5, 6, 7, 8, 14



   (3) 挑邊順序為(weight)：5, 7, 8, 4, 4, 6, 14, 3
   圖與上題相同
   (4) 兩者皆在 adjacency matrix 執行, time complexity : O($|V^2|$)

3.
   (1) 是一個 complete binary tree 且
   root 具最小值
   其 leftchild 與 rightchild 亦是一個 min-heap
   (2) 想法：由 root 開始往下 check, check 之原則為 root <= leftchild 且 root <= rightchild(假設題目給的 BT 為 complete BT)
   [algo]

```
void minHeapcheck( Node *root )
{
      if( root.left  ≠NULL )
            minHeapcheck( left );
      else if ( root.right  ≠NULL )
            minHeapcheck( right );
      else
      {
            if( root.value > left.value || root.value > right.value )
                  return “this is not a min-Heap”;
      }
}
```

4. F　F　T　F　F | T　F　F

5. (1) $\Theta(n\log^2 n)$
   (2) $\Theta(n\lg\lg n)$
   (3) $\Theta(\lg n* \lg\lg n)$
   (4) $\Theta(\lg n)$
   (5) $\Theta(\lg n)$

6.
   (1) X = 1, Y = 2, Z = 1;
   (2) I HAVE NO IDEA
   (3) max flow = 5
   min-cut = {s}{a, c, d, b, e, f, t}

7.
   (1) 140
   (2) step1: 對 A[1], …, A[n]作 sort

```
step2: i = 1; j = n;
     while ( i ≠ n )
     {
          B[i] <- A[i];
          B[i+1] <- A[j];
          i +=2;
          j = j – 1;
     }
  Time Complexity:
     Step1: O( nlgn )
     Step2: O( n )
     ➔ total : O( nlgn )
```

8.