

## Tutorial 4: Analysis of Algorithm

**Q1** The function `subset()` below takes two linked lists of integers and determines whether the first is a subset of the second. Give the worst-case running time of `subset` as a function of the lengths of the two lists. When will this worst case happen?

```

1  typedef struct _listnode{
2      int item;
3      struct _listnode *next;
4  } ListNode;
5
6  //Check whether integer X is an element of linked list Q
7  int element (int X, ListNode* Q)
8  {
9      int found; //Flag whether X has been found
10     found = 0;
11     while ( Q != NULL && !found) {
12         found = Q->item == X;
13         Q = Q->next;
14     }
15     return found;
16 }
17
18 // Check whether L is a subset of M
19 int subset (ListNode* L, ListNode* M)
20 {
21     int success; // Flag whether L is a subset so far
22     success = 1;
23     while ( L != NULL && success) {
24         success = element(L->item, M);
25         L = L->next;
26     }
27     return success;
28 }

```

**Q2** Find the number of `printf` used in the following functions. Write down its time complexity in  $\Theta$  notation in terms of  $N$ .

```

1  void Q2a (int N)
2  {
3      int j, k;
4      for (j=1; j<=N; j*=3)
5          for(k=1; k<=N; k*=2)
6              printf("SC1007\n");
7  }

```

$$\frac{N}{3} \times \frac{N}{2}$$

```
1 void Q2b (int N)
2 {
3     int i;
4     if (N > 0)
5     {
6         for (i = 0; i < N; i++)
7             printf("SC1007\n");
8         Q2b(N-1);
9         Q2b(N-1);
10    }
11 }
```

**Q3** A sequence,  $x_1, x_2, \dots, x_n$ , is said to be cyclically sorted if the smallest number in the sequence is  $x_i$  for some  $i$ , and the sequence,  $x_i, x_{i+1}, \dots, x_n, x_1, x_2, \dots, x_{i-1}$  is sorted in increasing order. Design an algorithm to find the minimal element in the sequence in  $\mathcal{O}(\log n)$  time. What is the worst-case scenario?

2