

Solution 2: Analysis of Algorithm and Stack and Queue

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 }
```

S1 Let $|L|$ and $|M|$ indicate the length of the linked lists, L and M , respectively. The worst-case running time of `subset`:

- the first $|L| - 1$ elements of L from the last $|L| - 1$ elements of M in reverse order.
- the last element of L not in M

\therefore Total number of comparisons between elements of L and M

$$\begin{aligned}
 &= |M| + (|M| - 1) + (|M| - 2) + \dots + (|M| - (|L| - 2)) + |M| \\
 &= |L||M| - (1 + 2 + 3 + \dots + (|L| - 2)) \\
 &= |L||M| - \frac{(|L| - 2)}{2}(1 + |L| - 2) \\
 &= |L||M| - \frac{(|L| - 2)(|L| - 1)}{2} \\
 &= \Theta(|L||M|)
 \end{aligned}$$

Here we assume that $|L| < |M|$

Q2 Find the number of printf used in the following functions. Write down its time complexity in Θ 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 }

```

```

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 }

```

S2a Let K denote the number of iterations for the inner loop and J denote the number of iteration for the outer loop.

For the inner loop, we have

$$\begin{aligned}
 2^{K-1} &\leq N < 2^K \\
 (K-1) &\leq \log_2 N < K \\
 K &\leq \log_2 N + 1 < K + 1 \\
 K &= \lfloor \log_2 N + 1 \rfloor = \lfloor \log_2 N \rfloor + 1
 \end{aligned}$$

For the outer loop, we have

$$\begin{aligned} 3^{J-1} &\leq N < 3^J \\ (J-1) &\leq \log_3 N < J \\ J &\leq \log_3 N + 1 < J + 1 \\ J &= \lfloor \log_3 N + 1 \rfloor = \lfloor \log_3 N \rfloor + 1 \end{aligned}$$

\therefore The number of printf is $JK = (\lfloor \log_3 N \rfloor + 1)(\lfloor \log_2 N \rfloor + 1) = \Theta((\log N)^2)$

S2b Let $W(N)$ denote the number of printf used in the function with problem size of N

$$\begin{aligned} W(N) &= 2W(N-1) + N \\ &= 2(2W(N-2) + (N-1)) + N \\ &= 2^2W(N-2) + 2(N-1) + N \\ &= 2^2(2W(N-3) + (N-2)) + 2(N-1) + N \\ &= \dots \\ &= 2^{N-1}(1) + 2^{N-2}(2) + \dots + 2^3(N-3) + 2^2(N-2) + 2(N-1) + N \\ &= \sum_{t=0}^{n-1} 2^t(n-t) \\ &= 2^{N+1} - 2 - N \end{aligned}$$

\therefore The number of printf is $\Theta(2^N)$

Q3 Given the precedence of some operators,

Operators	Precedence
*, /, %	highest
+, -	
<<, >>	
&&	
=	lowest

- (a) convert an infix expression, $x = a + b * c \% d >> e$, to a postfix expression
- (b) convert a prefix expression, $= y \&\& << ab >> c + de$, to an infix expression
- (c) convert a postfix expression, $abc * d \% + e >> =$, to a prefix expression

S3 (a) $abc * d \% + e >> =$

(b) $y = a << b \&\& c >> d + e$

(c) $= x >> + a \% * bcde$