

แบบฝึกหัด บทที่ 1 การวิเคราะห์และออกแบบอัลกอริทึม

1. กำหนดให้  $n$  เป็นขนาดของอินพุต จงประมาณเวลาโดยวิธีนับรอบการทำงานของ basic operation และเขียนในรูปของ  $T(n)$

1	<pre>sum = 0; for(i=2; i&lt;n+1; i++)     sum++;</pre>
2	<pre>sum=0; for(i=0; i&lt;n; i=i+2)     sum++;</pre>
3	<pre>sum=0; for(i=1; i&lt;n; i=i*2)     sum++;</pre>
4	<pre>sum=0; for(i=0; i&lt;n; i++)     for(j=0; j&lt;n*n; j++)         sum++;</pre>
5	<pre>sum=0; for(i=0; i&lt;n; i++)     for(j=0; j&lt;i; j++)         sum++;</pre>
6	<pre>sum=0; for(i=0; i&lt;n; i++)     for(j=0; j&lt;i*i; j++)         for(k=0; k&lt;j; k++)             sum++;</pre>
7	<pre>sum=0; for(i=0; i&lt;n; i++)     sum++; val =1; for(j=0; j&lt;n*n; j++)     val=val+n;</pre>
8	<pre>count=1; while(n&gt;1) { count+=1;   n =n/2; }</pre>
9	<pre>i = 0; j = n; while(i&lt;j) { i=i+3;</pre>

แบบฝึกหัด บทที่ 1 การวิเคราะห์และออกแบบอัลกอริทึม

	j=j-5; }
10	<pre> for(i=1; i&lt;=n; i++) {   for(j=1; j&lt;=n; j++)   {     sum++;     continue;   } } </pre>

2. จากอัลกอริทึมต่อไปนี้ จงหาขนาดของอินพุต (input size) พร้อมประมาณเวลาทำงานของอัลกอริทึม กรณีเลวร้ายสุด (worst case) กรณีดีที่สุด (best case) และกรณีเฉลี่ย (average case) ตามลำดับ

1	<pre> function reverseArray(arr[], start, end)   while start &lt; end   swap arr[start] and arr[end]   start=start +1   end=end -1   end while </pre>
2	<pre> function bubbleSort(A)   n =length(A)   for i from 1 to n -1     for j from 0 to n -i -1       if A[j]&gt; A[j +1]         swap A[j] and A[j +1] </pre>
3	<pre> function binarySearch(A[], l, r, k)   while l &lt;=r:     m =l +(r -l)/2     if (A[m]==k)       return m     if (A[m]&lt; k)       l =m +1     else       r =m -1   end while   return -1 </pre>

3. จากฟังก์ชันบิกโอ (Big-Oh) ในแต่ละข้อ จงเขียนให้อยู่ในรูปฟังก์ชันบิกโออย่างง่ายและใกล้เคียงที่สุด

ข้อ	ฟังก์ชัน Big-Oh	คำตอบ
3.1	$O(3n - 5)$	$O(n)$
3.2	$O(2n^3 + 4n)$	
3.2	$O(1/n + 1)$	
3.3	$O(7n^8 + 3n^2 - 4)$	

แบบฝึกหัด บทที่ 1 การวิเคราะห์และออกแบบอัลกอริทึม

3.4	$O(5n + 4n \log n)$	
3.5	$O(n + \log n + 8000)$	

4. กำหนดให้  $T(n) = 2n^3 + 4n$  แต่ละข้อต่อไปนี้ข้อใดกล่าวถูกต้อง

- (a)  $T(n) = O(n^3)$
- (b)  $T(n) = O(n^5)$
- (c)  $T(n) = O(n^3 + 5n^2)$
- (d)  $T(n) = O(n)$
- (e)  $T(n) = \Omega(n^3)$
- (f)  $T(n) = \Omega(n^2)$
- (g)  $T(n) = \Theta(n^3)$
- (h)  $T(n) = \Theta(n^4)$

ตอบ .....

5. จงพิสูจน์แต่ละข้อว่าเป็นจริงหรือเท็จ

5.1  $T(n) = 3n^3 + 2n + 7 \in \Theta(n^3)$

5.2  $T(n) = n^3 + 20n + 1 \notin O(n^2)$

5.3  $T(n) = n^3 + 20n \in \Omega(n^2)$

6. กำหนดให้  $n$  เป็นขนาดของอินพุต จงเรียงลำดับฟังก์ชันต่อไปนี้จากน้อยไปมาก

$5n^3 + 2n^2 + 30$	$7 \log n$	$10n$	$n^{100}$
$50000$	$0.5n^3$	$n^{1/3}$	$100 \cdot 2^n$

7. กำหนดให้อาร์เรย์จำนวนเต็ม  $A$  ขนาด  $n$  จำนวน จงออกแบบอัลกอริทึม  $n^2$  เพื่อหาว่ามีสมาชิกสองตัวที่ผลรวมเท่ากับ  $x$  เช่น  $A[] = \{5, 15, -30, 10, -5, 40, 10\}$  และ  $x = 20$  ซึ่งได้แก่  $\{5, 15\}$  และ  $\{10, 10\}$  ตามลำดับ

แบบฝึกหัด บทที่ 1 การวิเคราะห์และออกแบบอัลกอริทึม

8. กำหนดให้อาร์เรย์จำนวนเต็ม  $A$  ขนาด  $n$  จำนวน จงออกแบบอัลกอริทึม  $n^3$  เพื่อหาผลรวมของลำดับต่อเนื่องที่มากที่สุด (maximum contiguous subsequence) ตัวอย่างเช่น  $A[] = \{5, 15, -30, 10, -5, 40, 10\}$  ผลรวมของลำดับต่อเนื่องที่มากที่สุด คือ  $\{10, -5, 40, 10\}$  ซึ่งก็คือ  $10 - 5 + 40 + 10 = 55$