Lab01-AlgorithmAnalysis

CS214-Algorithm and Complexity, Xiaofeng Gao, Spring 2020.

∗ If there is any problem, please contact TA Shuodian Yu.

∗ Name:_____ Student ID:_____ Email: _____

1. Please analyze the time complexity of Alg. 1 with brief explanations.

Algorithm 1: PSUM

Input: $n = k^2$, k is a positive integer.
Output: $\sum_{i=1}^{j} i$ for each perfect square j between 1 and n.

```
1 k ← √n;
2 for j ← 1 to k do
3 |   sum[j] ← 0;
4 |   for i ← 1 to j^2 do
5 |   |   sum[j] ← sum[j] + i;
6 return sum[1 ··· k];
```

2. Analyze the **average** time complexity of QuickSort in Alg. 2.

Algorithm 2: QuickSort

Input: An array $A[1, \cdots, n]$
Output: $A[1, \cdots, n]$ sorted nondecreasingly

```
1 pivot ← A[n]; i ← 1;
2 for j ← 1 to n − 1 do
3 |   if A[j] < pivot then
4 |   |   swap A[i] and A[j];
5 |   |   i ← i + 1;
6 swap A[i] and A[n];
7 if i > 1 then QuickSort(A[1, ··· , i − 1]);
8 if i < n then QuickSort(A[i + 1, ··· , n]);
```

3. The BubbleSort mentioned in class can be improved by stopping in time if there are no swaps during an iteration. An indicator is used thereby to check whether the array is already sorted. Analyze the **average** and **best** time complexity of the improved BubbleSort in Alg. 3.

Algorithm 3: BubbleSort

Input: An array $A[1, \ldots, n]$
Output: $A[1, \ldots, n]$ sorted nondecreasingly

```
1 i ← 1;sorted ← false;
2 while i ≤ n − 1 and not sorted do
3 |   sorted ← true;
4 |   for j ← n downto i + 1 do
5 |   |   if A[j] < A[j − 1] then
6 |   |   |   interchange A[j] and A[j − 1];
7 |   |   |   sorted ← false;
8 |   i ← i + 1;
```

4. Rank the following functions by order of growth with brief explanations: that is, find an arrangement $g_1, g_2, \ldots, g_{15}$ of the functions $g_1 = \Omega(g_2), g_2 = \Omega(g_3), \ldots, g_{14} = \Omega(g_{15})$. Partition your list into equivalence classes such that functions $f(n)$ and $g(n)$ are in the same class if and only if $f(n) = \Theta(g(n))$. Use symbols "$=$" and "$\prec$" to order these functions appropriately. Here $\log n$ stands for $\log_2 n$.

$2^{\log n}$	$(\log n)^{\log n}$	n^2	$n!$	$(n+1)!$
2^n	n^3	$\log^2 n$	e^n	2^{2^n}
$\log \log n$	$n \cdot 2^n$	n	$\log n$	$4^{\log n}$

Remark: You need to include your .pdf and .tex files in your uploaded .rar or .zip file.