Lab01-Algorithm Analysis

CS214-Algorithm and Complexity, Xiaofeng Gao, Spring 2019.

∗ If there is any problem, please contact TA Mingran Peng. Also please use English in homework.

∗ Name: 田雪飞 Student ID: 515030910347 Email: 13487426939@qq.com

1. **Solution.** The following is solution process:

Algorithm	Time Complexity	Space Complexity
$InsertionSort$	$O(n^2)$, $\Omega(n)$	$\Theta(1)$
$CocktailSort$	$O(n^2)$,$\Omega(n)$	$\Theta(1)$
$SelectionSort$	$O(n^2)$,$\Omega(n)$	$\Theta(1)$

CocktailSort:
Line 4: The number of **comparisons** carried out by Cocktailsort in loop

$$n-1 \tag{1}$$

and the number of **swap and assignment operation** at at least(**best**)

$$0 \tag{2}$$

at most(**worst**)

$$2(n-1)+2(n-3)+\cdots+1 \tag{3}$$

Line 9: The number of **comparisons** carried out by in loop at most least(**best**)

$$n-2 \tag{4}$$

and the number of **swap and assignment operation** at least(**best**)

$$0 \tag{5}$$

at most(**worst**)

$$2(n-2)+2(n-4)+\cdots+0 \tag{6}$$

so, the number of total executed in the loop of while at least**(best)**

$$T(n)=n-1+n-2=2n-3 \tag{7}$$

at most **(worst)**

$$T(n)=n-1+n-2+2\sum_{0}^{n-1}k=n^2+n-3 \tag{8}$$

so,Time Complexity is $O(n^2)$ and $\Omega(n)$,and Space Complexity is $\Theta(1)$.

SelectionSort:
Line 2: The loop executed from 1 to $n-1$ **so**,the number of **assignment operations** carried out by SelectionSort are:

$$2(n-1) \tag{9}$$

Line 4: The comparisons of $A[j]>max$ will be executed when
$j=2$,the number of comparisons are $n-1$;
$j=3$,the number of comparisons are $n-2$;

$\vdots$
$j = n$,the number of comparisons are 0;
so,the number of **comparisons** carried out by SelectionSort are:

$$\sum_{0}^{n-1} j = \frac{n(n-1)}{2} \tag{10}$$

The number of **assignment operations** carried out in second loop are
at most:

$$n(n-1) \tag{11}$$

at least:

$$0 \tag{12}$$

The number of total executed in SelectionSort are at most**(worst)**:

$$T(n) = 2(n-1) + \frac{n(n-1)}{2} + n(n-1) = \frac{1}{2}(3n^2 + n - 4); \tag{13}$$

The number of total executed in SelectionSort are at most**(best)**:

$$T(n) = 2(n-1) + \frac{n(n-1)}{2} = \frac{1}{2}(n^2 + 3n - 4); \tag{14}$$

so,Time Complexity is $O(n^2)$ and $\Omega(n)$,and Space Complexity is $\Theta(1)$. □

2. **Solution.** The following is solution process:
   (a) Two stack to simulated queue:
   - **stack1** is used for **enqueue**,**stack2** is used for **dequeue**;
   - when pushing an element,we push it into **stack1**;
   - when popping an element,if **stack2** is empty,we push all elements in **stack1** into **stack2**,if **stack2** is not empty,we pop an element in **stack2** directly.

   (b) Time Complexity(potential function):
   Potential function: $\Phi(S)$ denote the num[i] of items in **stack1**.
   State: Here state S_i refers to the state of the **stack1** after the i-th.
   Correctness: $\Phi(S_i) \geq 0 = \Phi(S_0)$ for any i;
   According to the definition of $\Phi(S)$,we know

$$\Phi(S_i) = num[i] \tag{15}$$

PUSH: $\begin{aligned}\hat{C}_i &= C_i + \Phi(S_i) - \Phi(S_0)\\ &= 1 + num[i] - num[i-1]\\ &= 1 + num[i-1] + 1 - num[i-1]\\ &= 2\end{aligned}$

POP:
- if **stack2** is not empty:
$$\begin{aligned}\hat{C}_i &= C_i + \Phi(S_i) - \Phi(S_0)\\ &= 1 + num[i] - num[i-1]\\ &= 1\end{aligned}$$
- if **stack2** is empty:
$$\begin{aligned}\hat{C}_i &= C_i + \Phi(S_i) - \Phi(S_0)\\ &= num[i-1] + 1 + (num[i] - num[i-1])\\ &= num[i-1] + 1 + (0 - num[i-1])\\ &= 1\end{aligned}$$

Thus, starting from two empty stacks,any sequence of **n1 PUSH**,**n2 POP** operations takes at most

$$T(n) = \sum_{i}^{n} C_i \leq \sum_{i}^{n} \hat{C}_i = 2n_1 + n_2. \quad \textit{Here} \quad n = n_1 + n_2 \quad \textbf{and} \quad n_1 \geq n_2. \tag{16}$$

□