

Homework 3

Lei Zhang

February 23, 2016

1 Exercises 3.1 - 5

Algorithm 1: Identifying topologies

```
//Input: An boolean matrix  $A[0..n-1, 0..n-1]$ , where  $n > 3$ ;  
//Output: 0 denotes the topology is a ring, 1 denotes the topology is a star,  
2 denotes the topology is a fully connected mesh, 3 denotes the topology is  
none of the three choices;  
for  $i \leftarrow 0$  to  $n-1$  do  
     $sumOfLine[i] \leftarrow 0$   
end for  
for  $i \leftarrow 0$  to  $n-1$  do  
    for  $j \leftarrow 0$  to  $n-1$  do  
         $sumOfLine[i] \leftarrow sumOfLine[i] + A[i, j]$   
    end for  
end for  
for  $i \leftarrow 0$  to  $n-1$  do  
    if  $sumOfLine[i] == 2$  then  
         $countOf_2++ = 1$   
    end if  
end for  
for  $i \leftarrow 0$  to  $n-1$  do  
    if  $sumOfLine[i] == n-1$  then  
         $countOf_{nminus1}++ = 1$   
    end if  
end for  
if  $countOf_2 == n-1$  then  
    return 0  
end if  
if  $countOf_2 == n-2$  and  $countOf_{n-1} == 1$  then  
    return 1  
end if  
if  $countOf_{nminus1} == n-1$  then  
    return 2  
end if
```

return 3

Time efficiency is $\Theta(n)$

2 Exercises 3.1 - 7

3 Exercises 3.1 - 8

- outer loop 1: $min = 2$. The list is A, X, E, M, P, L, E
- outer loop 2: $min = 2$. The list is A, E, X, M, P, L, E
- outer loop 3: $min = 6$. The list is A, E, E, M, P, L, X
- outer loop 4: $min = 5$. The list is A, E, E, L, P, M, X
- outer loop 5: $min = 5$. The list is A, E, E, L, M, P, X
- outer loop 6: $min = 5$. The list is A, E, E, L, M, P, X

4 Exercises 3.1 - 11

- outer loop 1: The list is E, A, M, P, L, E, X
- outer loop 2: The list is A, E, M, L, E, P, X
- outer loop 3: The list is A, E, L, E, M, P, X
- outer loop 4: The list is A, E, E, L, M, P, X
- outer loop 5: The list is A, E, E, L, M, P, X
- outer loop 6: The list is A, E, E, L, M, P, X

5 Exercises 3.1 - 12

5.1 a.

Assume the list was not sorted before a outer loop, at least one element must be greater than it's former element. If so, at least one exchange must be made. Thus, if bubble sort makes no exchages on its pass through a list, the list must be sorted.

5.2 b.

Algorithm 2: BubbleSort

```
//Input: An array  $A[0..n-1]$  of orderable elements
//Output: Array  $A[0..n-1]$  sorted in nondecreasing order
numOfSwap  $\leftarrow 0$ 
for  $i \leftarrow 0$  to  $n-2$  do
  for  $j \leftarrow 0$  to  $n-2-i$  do
    if  $A[j+1] < A[j]$  then
      swap $A[j]$ and $A[j+1]$ 
      numOfSwap  $+= 1$ 
    end if
  end for
  if numOfSwap  $== 0$  then
    Break
  end if
end for
```

5.3 c.

It is known that the standard bubble sort has quadratic performance in the worst case. The worst case is to sort an array in increasing order. For this situation, bubble sort will exchange elements in every outer loop. The improvement won't get a change to take effect. Thus, worst-case efficiency of the improved version is still quadratic.

6 Exercises 3.4 - 10

6.1 a.

$$\sum_{i=0}^{n^2} i =$$

6.2 b.

6.3 c.

6.4 d.

6.5 e.

7 Exercises 3.5 - 1

8 Exercises 3.5 - 4