Wednesday 27 April 2022
14:00–15:30 BST
Duration: 1 hour 30 minutes
Additional time: 30 minutes
Timed exam – fixed start time

DEGREES OF MSci, MEng, BEng, BSc, MA and MA (Social Sciences)

ALGORITHMS AND DATA STRUCTURES 2
COMPSCI2007

Answer all 5 questions

This examination paper is an open book, online assessment and is worth a total of 60 marks

1. Algorithm **F** takes as input an array of integers A and two indices l, r of A. Assume A contains both negative and positive integers and no duplicate elements. The algorithm is described by the following pseudocode:

```
F(A, l, r)
   x := 0
   y := 0
   if l = r
      x := A[l]
      y := A[l]
   else if l + 1 = r
      if A[l] < A[r]
         x := A[r]
         y := A[l]
      else
         x := A[l]
         y := A[r]
   else
      q := l + (r − l)/2      // integer division
      (xl, yl) := F(A, l, q)
      (xr, yr) := F(A, q + 1, r)
      if xl > xr
         x := xl
      else
         x := xr
      if yl < yr
         y := yl
      else
         y := yr
   return (x, y)
```

(a) Briefly explain what algorithm **F** implements. [5]

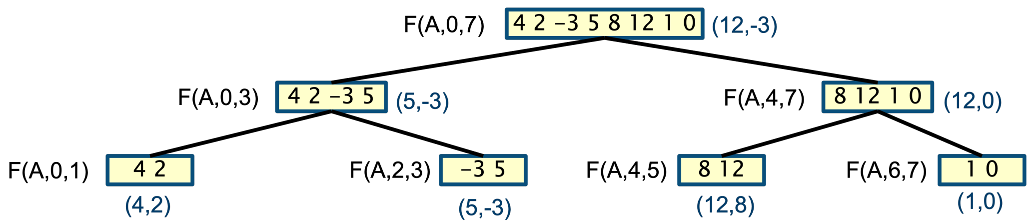
> **Solution:** This algorithm finds the maximum (1 point) and minimum (1 point) elements in the subarray $A[l..r]$ (1 point). The maximum and minimum values are stored in variables x and y (2 points), respectively.

(b) What is the output of $\mathbf{F}(A, 0, 7)$, where $A = [4, 2, -3, 5, 8, 12, 1, 0]$? [2]

> **Solution:** The output is the pair $(12, -3)$.

(c) Draw the recursion tree for $\mathbf{F}(A, 0, 7)$. [3]

Solution:



1 point for correct tree structure, 1 point for correct input values, and 1 point for correct return values.

(d) Is **F** an in-place algorithm? Justify your answer. [2]

Solution: Yes, as no auxiliary data structure is allocated during the execution of the algorithm. Only the following integer auxiliary variables are used: x, y, q, xl, yl, xr, and yr.

(e) Identify and briefly explain with simple language the Divide, Conquer and Combine steps in Algorithm **F**. [3]

Solution: The three steps are as follows:

Divide: input array A is divided into two equal parts around the mid-value.

Conquer: recursively find the maximum and minimum of both the left and right sub-arrays.

Combine: compare the maximum and minimum of both sub-arrays to find the maximum and minimum of the whole array.

(f) Write the recurrence equation for **F** and solve it to compute its running time using Big-O notation. [5]

Solution: The recurrence equation is

$$T(1) = T(2) = O(1)$$
$$T(n) = T(n/2) + T(n/2) + O(1)$$

Solving with the iterative method we obtain

$$\begin{aligned}
T(n) &= 2T(n/2) + c_1 \\
&= 4T(n/4) + 2c_1 \\
&= \dots \\
&= 2^k T(n/2^k) + kc_1 \\
&= 2^{\log n} T(1) + (\log n)c_1 \\
&= nc_2 + (\log n)c_1 \\
&= O(n)
\end{aligned}$$

1 point for base case, 1 point for step case, 1 point for correct iteration up to k, 1 point for correct substitution with $\log n$, 1 point for correct solution.

2. Apply the Master Theorem to give tight asymptotic bounds for the following recurrences. In your answers, briefly explain which case applies and why.

(a) $T(n) = 2T(n/2) + n^3$ [3]

Solution: This is a divide-and-conquer recurrence with $a = 2$, $b = 2$, $f(n) = n^3$, and $c = 3$. Since $c > \log_b a$, that is $3 > \log_2 2 = 1$, case 3 of the master theorem applies, and $T(n) = \Theta(n^3)$.

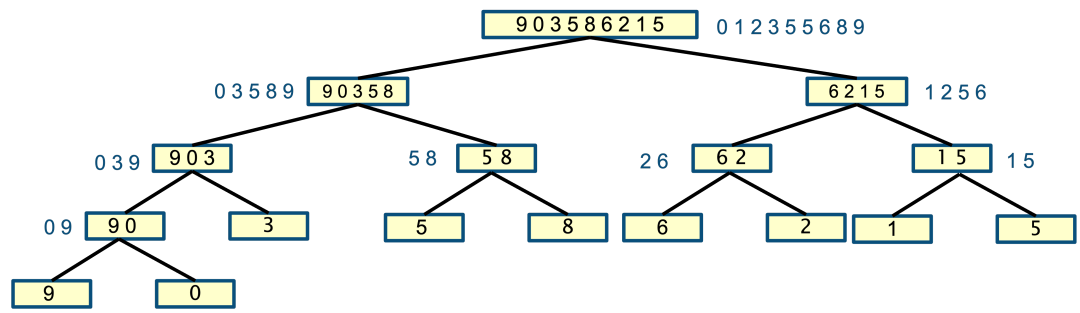
(b) $T(n) = 16T(n/4) + n^2$ [3]

Solution: This is another divide-and-conquer recurrence with $a = 16$, $b = 4$, $f(n) = n^2$, and $c = 2$. Since $c = \log_b a$, that is $2 = \log_4 16 = 2\log_4 4 = 2$, case 2 of the master theorem applies, and $T(n) = \Theta(n^2 \log n)$.

3. Draw the recursion tree computed when sorting array $A = [9, 0, 3, 5, 8, 6, 2, 1, 5]$ with the following two algorithms:

(a) **MERGE-SORT**$(A, 0, 8)$. [5]

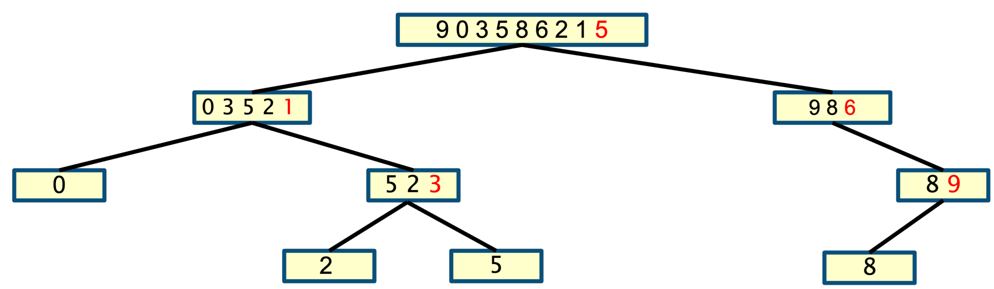
Solution:



3 points for correct tree structure, 2 point for partial solutions.

(b) **QUICKSORT**$(A, 0, 8)$. Assume the right-most element is selected as pivot when partitioning. [5]

Solution:



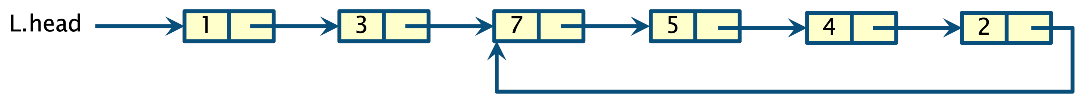
3 points for correct tree structure, 2 points for correct pivot selections.

4. **(a)** Consider a bounded queue implemented by array $Q[0..4]$. Illustrate the result of each operation in the sequence **ENQUEUE**$(Q, 2)$, **DEQUEUE**(Q), **ENQUEUE**$(Q, 6)$, **ENQUEUE**$(Q, 0)$, and **DEQUEUE**(Q). Assume the initial configuration is $Q = [*, 1, 7, 5, *]$, with $Q.head = 1$ and $Q.tail = 4$. [5]

Solution:

- **ENQUEUE**$(Q, 2)$, $Q = [*, 1, 7, 5, 2]$, $Q.head = 1$, $Q.tail = 0$;
- **DEQUEUE**(Q), return 1, $Q = [*, *, 7, 5, 2]$, $Q.head = 2$;
- **ENQUEUE**$(Q, 6)$, $Q = [6, *, 7, 5, 2]$, $Q.head = 2$, $Q.tail = 1$;
- **ENQUEUE**$(Q, 0)$, error "overflow";
- **DEQUEUE**(Q), return 7, $Q = [6, *, *, 5, 2]$, $Q.head = 3$.

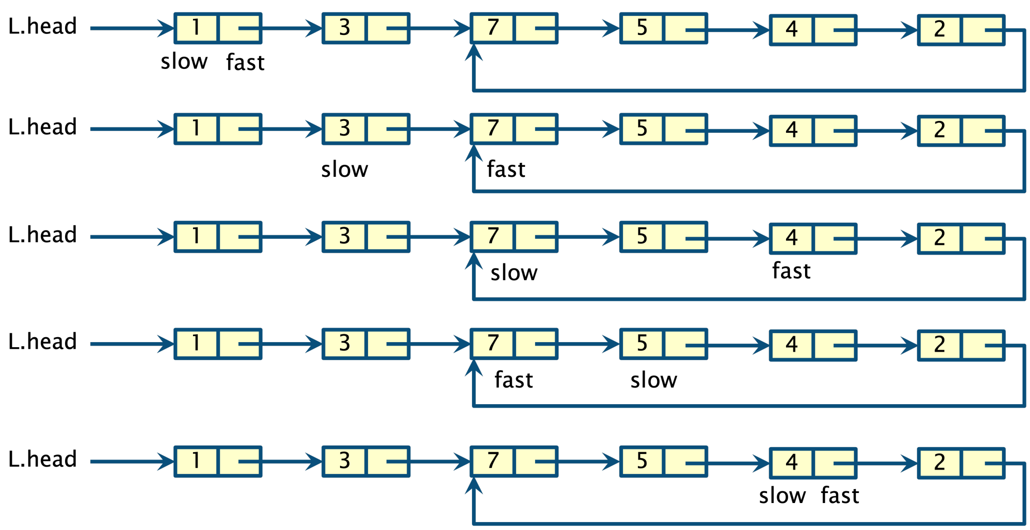
(b) Given the head of a singly-linked list, define using pseudocode an algorithm that detects if the list has a cycle. Use an iterator with slow and fast pointers. Illustrate how it operates on the following input:



[5]

Solution:

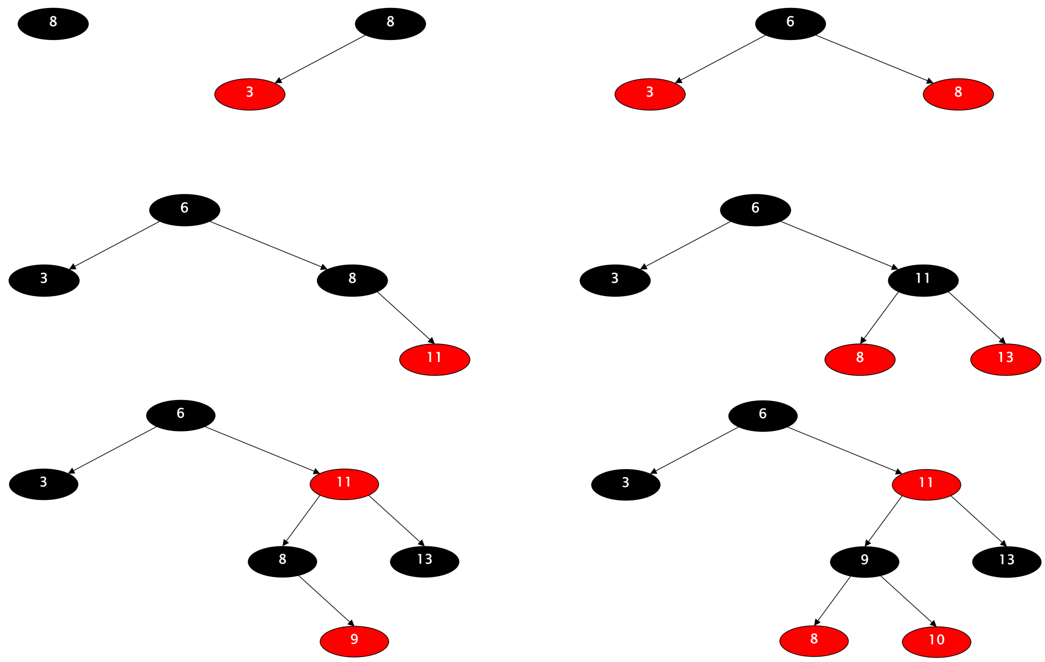
HAS-LOOP(l)
    $slow := l$
    $fast := l$
    while $fast \neq$ NIL and $fast.next \neq$ NIL
        $slow := slow.next$
        $fast := fast.next.next$
        if $slow = fast$
            return $true$
    return $false$



The output of **HAS-LOOP**$(L.head)$ is true.

5. **(a)** Insert keys 8,3,6,11,13,9,10 (in this order) in the empty red-black tree T and show at each step how T is updated. Explicitly mention at each step which operations are performed (e.g. rotations, recolourings, etc). [7]

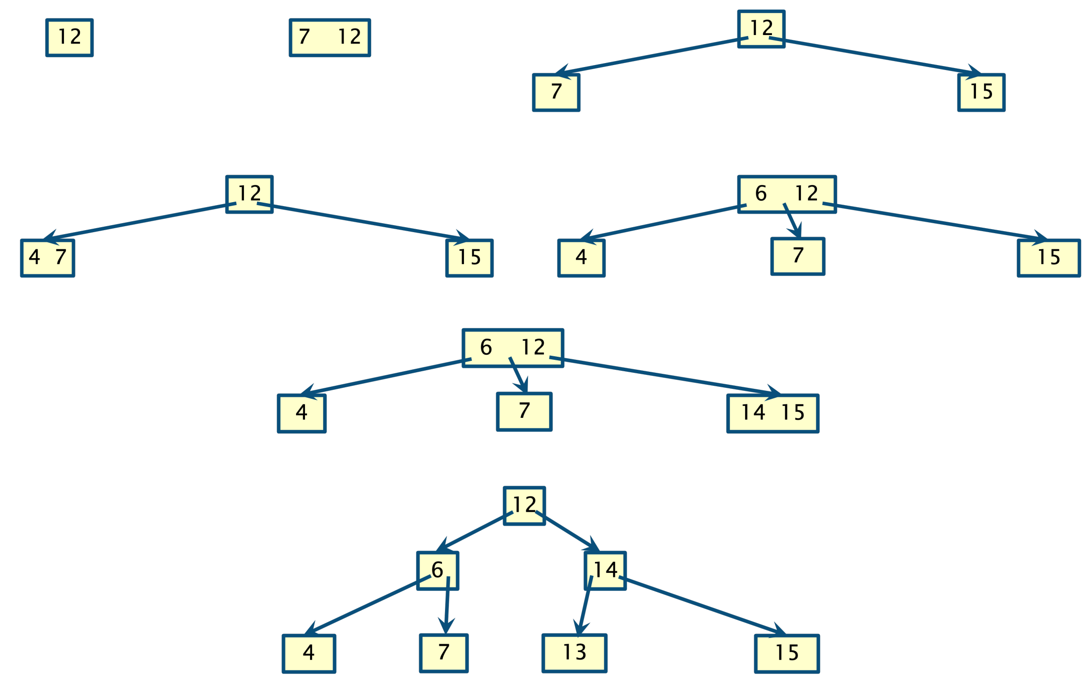
Solution:



1. Fix root
2. –
3. Left-rotation on 3, right-rotation on 6, recolouring 6 and 8
4. Recolouring 3 and 8, fix root
5. Left-rotation on 8, recolouring 11 and 8
6. Recolouring 8, 13, and 11
7. Left-rotation on 8, recolouring 8 and 9

(b) Insert keys 12,7,15,4,6,14,13 (in this order) in the empty B-tree T with $t = 2$ (i.e. each node can store at most 3 keys) and show at each step how T is updated. Explicitly mention at each step which operations are performed. Assume insertion implements a one-pass strategy. [7]

Solution:



1 point for each correct configuration and its explanation.

END OF QUESTION PAPER