Data Structures CSC 212 **Final Exam - Spring 2022**

Date: 04/06/2022 Duration: 3 hours

Student ID: Name:

1	2	3.1	3.2	4	5	6	7	8	Total

1.1 ___
1.2 ___
1.3 ___
1.4 ___
1.5 ___
1.6 ___

2.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.5 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.6 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.7 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.8 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.9 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.10 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.11 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2.12 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

3.a.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.a.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.a.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.a.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.a.5 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.5 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.6 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.7 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.8 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.9 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.10 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.11 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3.b.12 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

4.a.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
4.a.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
4.a.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ Ⓕ
4.a.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
4.b Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
4.c Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
4.d Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

5.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
5.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
5.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
5.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
5.5 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
5.6 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
5.7 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

6.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
6.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
6.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
6.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
6.5 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

7.a.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
7.a.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
7.a.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
7.b.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
7.b.2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
7.b.3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
7.b.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

8.1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
8.2.a ___
8.2.b ___
8.2.c ___
8.2.d ___
8.3.a ___
8.3.b ___
8.3.c ___
8.3.d ___
8.4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
8.5 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
8.6 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

King Saud University

College of Computer and Information Sciences

Department of Computer Science

Data Structures CSC 212

Final Exam - Spring 2022

Date: 04/06/2022

Duration: 3 hours

Question 1 .... 12 points

Choose the most appropriate data structure for each of the following tasks.

A. LinkedList	B. HashTable	C. DoubleLinkedList	D. LinkedQueue	E. BT	F. ArrayStack
G. ArrayQueue	H. BST	I. LinkedPQueue	J. BPlusTree	K. HeapPQueue	L. Graph.

1. A text editor supporting multi-step undo/redo functionality. ___
2. Processing a large dataset whose size is larger than the size of main memory. ___
3. Check if a string is a palindrome (reads the same in both directions, such as *level* or *radar*). ___
4. Validating users' passwords without needing to store them in clear text. ___
5. Organizing categories of products in an e-commerce website. ___
6. Navigating a robot vacuum cleaner around a house. ___

Question 2 .... 12 points

Write the method `arrangeList` which receives a list of integers l and a number n. The method should return a new list containing all the integers in l that are greater than or equal to n first followed by all the integers in l that are less than n. The list l must not change at the end of the method.

```java
public static LinkedList<Integer> arrangeList(LinkedList<Integer> l, int n) {
  List<Integer> x = new LinkedList<Integer>();
  List<Integer> y = new LinkedList<Integer>();
  if (...)
    System.out.println("List␣is␣empty");
  else {
    ...
    while (...) {
      if(l.retrieve() >= n)
        x.insert(l.retrieve());
      else
        ...
      ...
    }
    if(...)
      x.insert(l.retrieve());
    else
      y.insert(l.retrieve());
    if (...) {
      ...
      while (...) {
        x.insert(y.retrieve());
        ...
      }
      ...
    }
  }
  return ...
}
```

Example 1. *if List l: [1 5 8 4 3 6 7 2 9] and* $n = 5$.

Then, the method should return the list: [5 8 6 7 9 1 4 3 2]

1. Line 4:

   Ⓐ `l.hasNext()`

   Ⓑ `l.empty()`

   Ⓒ `l.full()`

   Ⓓ `l.last()`

   Ⓔ None

2. Line 7:

   Ⓐ `y.findNext();`

   Ⓑ `x.findFirst();`

   Ⓒ `l.findNext();`

   Ⓓ `l.findFirst();`

   Ⓔ None

3. Line 8:

   Ⓐ `l.retrieve()>= n`

   Ⓑ `l.retrieve()!= n`

   Ⓒ `l.last()`

   Ⓓ `!l.last()`

   Ⓔ None

4. Line 12:

   Ⓐ `l.findNext();`

   Ⓑ `y.insert(l.retrieve());`

   Ⓒ `y.insert(x.retrieve());`

   Ⓓ `l.retrieve();`

   Ⓔ None

5. Line 13:

   Ⓐ `x.findNext();`

   Ⓑ `l.findNext();`

   Ⓒ `x.remove();`

   Ⓓ `l.remove();`

   Ⓔ None

6. Line 15:

   Ⓐ `l.retrieve()== n`

   Ⓑ `l.retrieve()>=n`

   Ⓒ `x.retrieve()>=n;`

   Ⓓ `l.retrieve()!=n;`

   Ⓔ None

7. Line 19:

   Ⓐ `!x.empty()`

   Ⓑ `x.retrieve()!=` **null**

   Ⓒ `y.empty()`

   Ⓓ `!y.empty()`

   Ⓔ None

8. Line 20:

   Ⓐ `x.findFirst();`

   Ⓑ `y.findFirst();`

   Ⓒ `l.findNext();`

   Ⓓ `l.findFirst();`

   Ⓔ None

9. Line 21:

   Ⓐ `l.last()`

   Ⓑ `l.retrieve()!=` **null**

   Ⓒ `!y.last()`

   Ⓓ `!x.last()`

   Ⓔ None

10. Line 23:

    Ⓐ `l.retrieve();`

    Ⓑ `y.retrieve();`

    Ⓒ `y.findNext();`

    Ⓓ `l.findNext();`

    Ⓔ None

11. Line 25:

    Ⓐ `y.insert(l.retrieve());`

    Ⓑ `x.insert(l.retrieve());`

    Ⓒ `y.insert(x.retrieve());`

    Ⓓ `x.insert(y.retrieve());`

    Ⓔ None

12. Line 28:

Ⓐ `l;`

Ⓑ `x;`

Ⓒ `l.retrieve();`

Ⓓ `y;`

Ⓔ None

Question 3 .... 14 points

(a) Write the method `private int f(BTNode<T> t, T e, int k)`, member of `BT`, which returns the number of nodes in the level `k` of the subtree `t` having data equal to `e`. The root of the subtree (`t`) is at level 0.

```
1 private int f(BTNode<T> t, T e, int k) {
2   if(...)
3     return ...;
4   if (...)
5     return ...;
6   return ...;
7 }
```

1. Line 2:

Ⓐ `if (t.data == e)`

Ⓑ `if (t != null)`

Ⓒ `if (t == null)`

Ⓓ `if (e.equals(t.data))`

Ⓔ None

2. Line 3:

Ⓐ `return 0;`

Ⓑ `return k;`

Ⓒ `return 1;`

Ⓓ `return f(t.left,e,k)&&f(t.right,e,k);`

Ⓔ None

3. Line 4:

Ⓐ `if (!e.equals(t.data))`

Ⓑ `if (k == 0 && e.equals(t.data))`

Ⓒ `if (e.equals(t.data))`

Ⓓ `if (k == 1 && e.equals(t.data))`

Ⓔ None

4. Line 5:

Ⓐ `return f(t.left,e,k)+f(t.right,e,k);`

Ⓑ `return 0;`

Ⓒ `return 1;`

Ⓓ `return 1+f(t.left,e,k)+f(t.right,e,k);`

Ⓔ None

5. Line 6:

Ⓐ `return f(t.left,e,k)+f(t.right,e,k);`

Ⓑ `return f(t.left,e,k-1)+f(t.right,e,k-1);`

Ⓒ `return f(t,e,k+1)+f(t,e,k+1);`

Ⓓ `return f(t.left,e,k+1)+f(t.right,e,k+1);`

Ⓔ None

(b) Repeat the same questions as above, but this time as a user.

```
1  public static <T> int f(BT<T> b, T e, int k) {
2    if (...)
3      ...;
4    ...;
5    ...;
6  }
7  private static <T> int rf(BT<T> b, T e, int k) {
8    if (...)
9      return ...;
10   int n=0;
11   if (...) {
12     ...;
13     ...;
14   }
15 |  if (...) {
16 |     ...;
17 |     ...;
18 |  }
19 | }
```

1. Line 2:

   (A) `if (b.full())`

   (B) `if (b.empty())`

   (C) `if (!b.empty())`

   (D) `if (e.equals(b.retrieve()))`

   (E) None

2. Line 3: (A) `return 1;`

   (B) `return rf(b,e,k);`

   (C) `return 0;`

   (D) `return k;`

   (E) None

3. Line 4:

   (A) `return rf(b,e,k);`

   (B) `return e.equals(b.retrieve());`

   (C) `b.find(relative.Root);`

   (D) `b.find(relative.Parent);`

   (E) None

4. Line 5:

   (A) `return rf(b, e, k - 1);`

   (B) `return rf(b,e,k+1);`

   (C) `return rf(b,e,k);`

   (D) `return rf(b,e,0);`

   (E) None

5. Line 8:

   (A) `if (k==1)`

   (B) `if (e.equals(b.retrieve()))`

   (C) `if (k==1 && e.equals(b.retrieve()))`

   (D) `if (k==0 && e.equals(b.retrieve()))`

   (E) None

6. Line 9:

   (A) `return 0;`

   (B) `return 1+rf(b.left,e,k)+rf(b.right,e,k);`

   (C) `return 1;`

   (D) `return e.equals(b.retrieve());`

   (E) None

7. Line 11:

   (A) `if (b.find(Relative.Parent)){`

   (B) `if (b.find(Relative.LeftChild)){`

   (C) `if (b.find(Relative.Root)){`

   (D) `if (b.left != null){`

   (E) None

8. Line 12:

   (A) `n-=rf(b.left,e,k);`

   (B) `n-=rf(b,e,k);`

   (C) `n+=rf(b,e,k-1);`

   (D) `n+=rf(b,e,k+1);`

   (E) None

9. Line 13:

   (A) `b.find(Relative.RightChild);`

   (B) `b.find(Relative.Parent);`

   (C) `b.find(Relative.Root);`

   (D) `return n+1;`

   (E) None

10. Line 15:

    (A) `if (b.right != null){`

    (B) `if (b.find(Relative.RightChild)){`

    (C) `if (b.find(Relative.Parent)){`

    (D) `if (b.find(Relative.Root)){`

    (E) None

11. Line 16:

    (A) `n+=rf(b,e,k+1);`

    (B) `n-=rf(b,e,k);`

    (C) `n-=rf(b.right,e,k);`

    (D) `n+=rf(b,e,k-1);`

Ⓔ None

12. Line 17:

Ⓐ `b.find(Relative.Parent);`

Ⓑ `b.find(Relative.LeftChild);`

Ⓒ `return n+1;`

Ⓓ `b.find(Relative.Root);`

Ⓔ None

Question 4 . . . . 14 points

(a) Consider the following heap represented as an array: 2, 7, 5, 8, 20, 10, 12. Choose the correct answer for every operation (all operations are done on the above heap).

1. Heap after inserting 1: Ⓐ 1, 2, 5, 7, 20, 10, 12, 8 Ⓑ 1, 2, 5, 7, 20, 10, 8, 12 Ⓒ 2, 5, 7, 20, 10, 12, 8, 1 Ⓓ 2, 5, 7, 20, 10, 12, 1, 8 Ⓔ None
2. Heap after inserting 3 then 4: Ⓐ 2, 3, 4, 5, 20, 10, 12, 8, 7 Ⓑ 2, 3, 5, 4, 20, 10, 12, 8, 7 Ⓒ 2, 3, 4, 5, 8, 7, 20, 10, 12 Ⓓ 2, 3, 4, 5, 8, 10, 12, 7, 20 Ⓔ None
3. Heap after inserting 11 then deleting one key: Ⓐ 11, 2, 7, 5, 8, 20, 10, 12 Ⓑ 5, 3, 4, 20, 10, 12, 8, 7 Ⓒ 5, 7, 11, 8, 20, 10, 12 Ⓓ 5, 7, 10, 8, 20, 12, 11 Ⓔ None
4. Heap after deleting two keys: Ⓐ 2, 7, 5, 8, 20 Ⓑ 2, 5, 7, 20, 8 Ⓒ 7, 8, 10, 12, 20 Ⓓ 7, 10, 8, 12, 20 Ⓔ None

(b) Suppose we have two heaps (5, 9, 6) and (7, 8, 10) represented as arrays and a key 12, what will be the resultant heap after merging them? Ⓐ 12, 5, 9, 6, 7, 8, 10 Ⓑ 5, 9, 6, 12, 7, 8, 10 Ⓒ 5, 6, 7, 9, 12, 8, 10 Ⓓ 5, 9, 6, 7, 8, 10, 12 Ⓔ None

(c) What is the result of a bottom-up min-heap construction of the array 5, 11, 2, 7, 16, 15, 4? Ⓐ 2, 7, 4, 11, 16, 15, 5 Ⓑ 5, 11, 2, 7, 16, 15, 4 Ⓒ 2, 7, 5, 11, 16, 15, 4 Ⓓ 2, 4, 7, 11, 16, 15, 5 Ⓔ None

(d) What is the height of a heap containing 10 elements (an empty heap has height 0)? Ⓐ 3 Ⓑ 10 Ⓒ 4 Ⓓ 5 Ⓔ None.

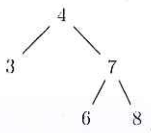
Question 5 . . . . 14 points

Choose the most appropriate answer:

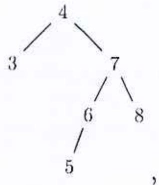
Remark 1. *Follow the the conventions: The height of an empty tree is 0. When necessary, replace with the smallest key in the right sub-tree.*

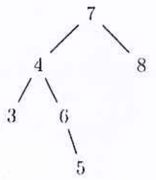
1. Which of the following is true for the best case run time for findkey in AVL trees: Ⓐ O(log n). Ⓑ O(n). Ⓒ O(1). Ⓓ O(n log n). Ⓔ None.
2. Which of the following is true for the average case run time for findkey in BST trees: Ⓐ O(log n). Ⓑ O(n). Ⓒ O(1). Ⓓ O(n log n). Ⓔ None.
3. The cost of one rotation in an AVL tree is:

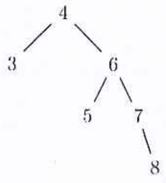
   Ⓐ $O(1)$. Ⓑ $O(\log n)$. Ⓒ $O(n)$. Ⓓ $O(n \log n)$ Ⓔ $O(n^2)$.

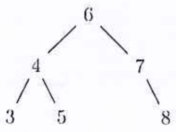
4. After inserting the key 5 in the AVL 

, the tree becomes:

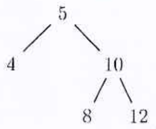
Ⓐ 

, Ⓑ 

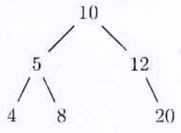
Ⓒ 

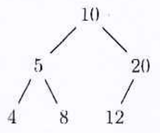
, Ⓓ 

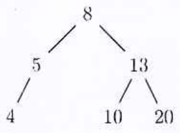
Ⓔ None

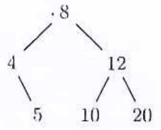
5. After inserting the key 20 in the AVL 

, the tree becomes:

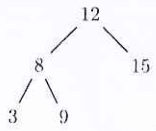
Ⓐ 

Ⓑ 

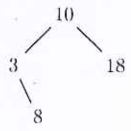
Ⓒ 

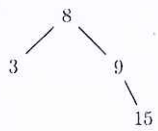
Ⓓ 

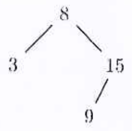
Ⓔ None

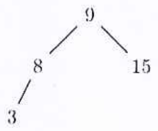
6. After deleting the key 12 from the AVL 

, the tree becomes:

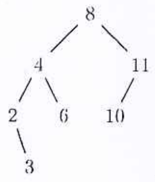
Ⓐ 

Ⓑ 

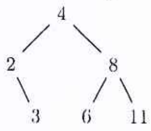
Ⓒ 

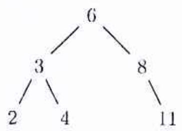
Ⓓ 

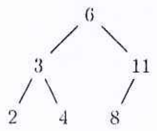
Ⓔ None

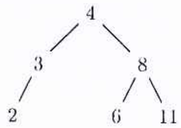
7. After deleting the key 10 from the AVL 

, the tree becomes:

Ⓐ 

Ⓑ 

Ⓒ 

Ⓓ 

Ⓔ None

Question 6 .... 14 points

Choose the most appropriate answer:

1. What is the average retrieval time when n keys hash to the same position? Ⓐ $O(n)$ Ⓑ $O(\log n)$ Ⓒ $O(1)$ Ⓓ $O(n^2)$ Ⓔ None

2. The best case running time for finding a key in a hash table with external chaining is: Ⓐ $O(1)$ Ⓑ $O(\log n)$ Ⓒ $O(n \log n)$ Ⓓ $O(n)$ Ⓔ None

3. Consider a hash table of size 7, with starting index zero, and a hash function $(3k + 4)\%7$. What is the content of the table when the sequence 1, 3, 8, 10 is inserted into the table using linear rehashing with $c = 1$? Note that '·' denotes an empty location. Ⓐ 8, ·, ·, ·, ·, ·, 10 Ⓑ 1, 8, 10, ·, ·, ·, 3 Ⓒ 1, ·, ·, ·, ·, ·,3 Ⓓ 1, 10, 8, ·, ·, ·, 3 Ⓔ None

4. Given the following input (4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199) and the hash function $k\%10$, which of the following statements are true? (1) 9679, 1989, 4199 hash to the same value. (2) 1471,

6171 hash to the same value. (3) All elements hash to the same value. (4) Each element hashes to a different value. Ⓐ (1) only Ⓑ (2) only Ⓒ (1) and (2) only Ⓓ (3) and (4) Ⓔ None

5. Consider the following hash function: select the two rightmost digits then apply mod 7 on the corresponding number. Which of the following couples of keys cause a collision?

   Ⓐ 3848 and 4756 Ⓑ 3973 and 1258 Ⓒ 162 and 35476 Ⓓ All of the above. Ⓔ None of the above.

Question 7 . . . . 14 points

(a) Choose the most appropriate answer:

Remark 2. *A B+ tree has two parameters, m: the maximum number of children and l: the maximum number of elements in a leaf node.*

1. In a B+ tree with $l = m = 6$, the maximum number of keys at level 1 (root at level 0):

   Ⓐ 29 keys Ⓑ 30 keys Ⓒ 5 keys Ⓓ 6 keys Ⓔ None

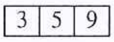
2. Consider a B+-tree in which the maximum number of keys in a non-leaf node is 5. What is the minimum number of keys in any non-root node?

   Ⓐ 2 Ⓑ 1 Ⓒ 4 Ⓓ 3 Ⓔ None

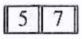
3. B+-trees and AVL trees have the same worst case time complexity for insertion and deletion:

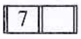
   Ⓐ *True* Ⓑ *False*

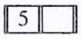
(b) Choose the correct result in each of the following cases (when possible, always borrow and transfer to the left). The order of the tree is $m = 3$:

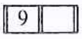
1. After inserting the key 7 in the B+ tree 

, the **root** of tree becomes:

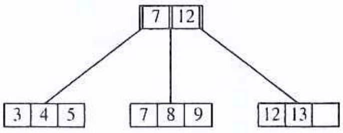
   Ⓐ 

Ⓑ 

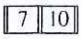
Ⓒ 

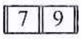
Ⓓ 

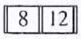
Ⓔ None

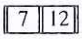
2. After inserting the key 10 in the B+ tree 

, the **root** of the tree becomes:

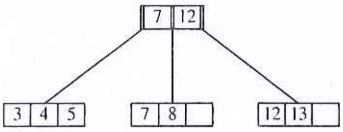
   Ⓐ 

Ⓑ 

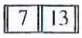
Ⓒ 

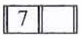
Ⓓ 

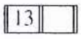
Ⓔ None

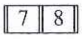
3. After deleting the key 12 from the B+ tree 

, the **root** of the tree becomes:

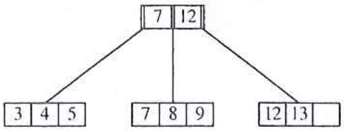
   Ⓐ 

Ⓑ 

Ⓒ 

Ⓓ 

Ⓔ None

4. After deleting the key 12 from the B+ tree 

, the **root** of the tree

becomes:

Ⓐ $\boxed{7}\boxed{}$ Ⓑ $\boxed{5}\boxed{}$ Ⓒ $\boxed{7}\boxed{9}$ Ⓓ $\boxed{7}\boxed{13}$ Ⓔ None

Question 8 . . . . 6 points

Given the following graph (G) adjacency matrix, answer the questions below.

	A	B	C	D	E	F
A	0	0	0	0	1	0
B	0	0	1	1	0	1
C	0	1	0	1	0	0
D	0	1	1	0	0	0
E	1	0	0	0	0	0
F	0	1	0	0	0	0

1. The total number of edges in G is: Ⓐ 3 Ⓑ 5 Ⓒ 7 Ⓓ 10 Ⓔ None.
2. Which of the following sequences are paths in this graph? Answer by T (true) or F (false).
   (a) (B, C, D, B, F) ___
   (b) (D, B, C, F) ___
   (c) (B, D, C, F, A) ___
   (d) (C, D, B, F) ___
3. Answer by T (true) or F (false).
   (a) G contains only two connected subgraphs. ___
   (b) There are n-1 edges in G, where n is the number of nodes. ___
   (c) G is a tree. ___
   (d) G is a connected graph. ___
4. Which of the following is true for this graph?
   Ⓐ The graph is connected. Ⓑ The graph has no cycles. Ⓒ The graph is not connected and has no cycles. Ⓓ The graph has cycles. Ⓔ None.
5. In terms of graph representation and space requirement, which statement is true about G?
   Ⓐ Adjacency list is more efficient than adjacency matrix. Ⓑ Adjacency matrix is more efficient than adjacency list. Ⓒ Both representations require the same space for G. Ⓓ Both representations require no space. Ⓔ None.
6. The number of cycles in this graph is:
   Ⓐ 0. Ⓑ 1. Ⓒ 2. Ⓓ 3. Ⓔ None.