Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination
Year 2, Semester 2 (2018)

IT2070-Data Structures and Algorithms

Duration: 2 Hours

October 2018

Instructions to Candidates:

- This paper is preceded by 10 minutes reading period. The supervisor will indicate when answering may commence.
- This paper has 4 questions.
- Answer questions in the booklet given.
- The total marks for the paper is 100.
- This paper contains 7 pages, including the cover page.
- Electronic devices capable of storing and retrieving text and mobile phones are not allowed.

Question 1 **(25 marks)**

a) Draw the resultant stack frame after performing the given operations to an empty stack s1 of size 4. (5 marks)

```
1) s1.push(6)
2) s1.push(10)
3) s1.push(11)
4) s1.push(s1.pop())
5) s1.push(1)
6) s1.peek()
```

b) i) A stack class called StackX has been created to store characters and 'push' and 'pop' methods have been implemented. Implement the peek method using push and pop methods. (4 marks)

ii) What is the time complexity of the peek method written in part b i) in Big O Notation ? (2 marks)

c) Complete the following five lines of the insert method of a circular queue class. (4 marks)

```
public void insert(int j) {
        if ..................... //line 1
            System.out.println("Queue is full");
        else {
             .....................   //line 2
             .....................   //line 3
             .....................   //line 4
             .....................   //line 5
             queArray[rear] = j;
             nItems++;
        }
}
```

d) *"rear = maxsize -1" can be used to find out whether a circular queue is full*

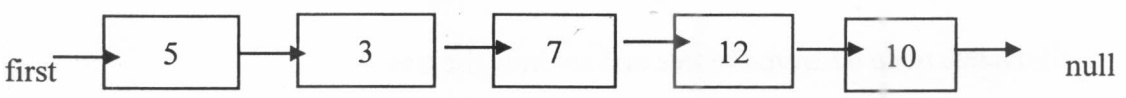
Do you agree with this statement? Justify your answer using a diagram (3 marks)

e) i) Explain how you would implement a queue using a linked list. (4 marks)

ii) Discuss one advantage of a queue implemented using a linked list over a queue implemented using an array. (3 marks)

Question 2 **(25 marks)**

a) Consider the following linked list.



Draw a diagram of the above linked list after the following lines of code have been executed. (2 marks)

```
Link temp = first.next;
Link nodeToInsert = new Link(4);
nodeToInsert.next = temp.next;
temp.next = nodeToInsert;
```

b) A linked list is used to store some employee data namely employee number (integer) and name (string).

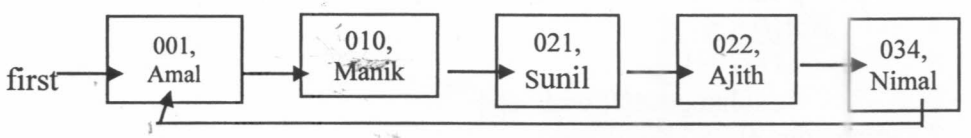
i) Implement **Link** class for the above linked list with suitable attributes and methods. (2 marks)

ii) Implement **Linked List** class with suitable attributes and the constructor. (1 mark)

iii) Write `sortedInsert(int empNo, String name)` method in **Linked List** class to insert the data into the linked list in ascending order. (6 marks)

iv) Write `displayList( )` method in **Linked List** class to display all the links in the linked list. (2 marks)

v) A singly circular linked list is a linked list where all the nodes are connected to each other. In circular linked list, the last link points to the first link as shown below.



Modify the `displayLink( )` method implemented in part b) iv) to print all the elements in the singly circular linked list. (3 marks)

c) i) Insert the following numbers to a binary search tree. (2 marks)

56 34 23 67 38 88 60 58 59 25

ii) Delete node 67 from the tree and re-draw the tree (2 marks)

iii) Describe the type of the tree you get, if you first apply quick sort algorithm to the numbers given above and then create the tree? (2 marks)

iv) Discuss the drawback of the tree you get in part c) iii) , if there are any. (3 marks)

Question 3 **(25 marks)**

a) Find the ***Big O*** value for the following functions. Justify your answer.

a) $T(n) = n + n\log n + 100$

b) $T(n) = n^8 + n + 2^n + 1$ (2 marks)

b) Compare and contrast the Merge sort and Quick sort algorithms. (2 marks)

c) Given below is an algorithm for ***QUICKSORT*** *(A, p, r)*

```
QUICKSORT (A, p, r)
1  if p < r
2      q = PARTITION(A, p, r)
3      QUICKSORT (A, p, q-1)
4      QUICKSORT (A, q+1, r)

PARTITION(A, p, r)
1  x = A[r]
2  i = p - 1
3  for j = p to r - 1
4      if A[j] ≤ x
5          i = i + 1
6          exchange A[i] with A[j]
7  exchange A[i + 1] with A[r]
8  return i + 1
```

i) Modify the above ***QUICKSORT*** *(A, p, r)* algorithm to sort the numbers in descending order. (Only the modified line should be described) (2 marks)

ii) Write a recurrence equation that describes the best case running time of quick sort algorithm. (4 marks)

d) Briefly describe the following terms as related to tree terminology.

i) Complete Binary tree
ii) Leaf Node

(3 marks)

e) Relationship between nodes (n) and height (h) of **Full Binary Tree** is given by $n = 2^{h+1} - 1$ Illustrate this using a diagram. (2 marks)

f) Illustrate the operations of the **Heap sort** for the array A of elements given below. (For the purpose of illustration, assign the values only once to the given algorithms at the end of the paper and use diagrammatic way to reach the answer.)

	1	2	3	4	5	6	7	8
A	25	15	35	55	5	75	65	45

(5 marks)

g) Consider the following **min-heap** A = <2, 7, 11, 16, 14, 12, 25, 40, 65>. Assume that the key 16 in the **min-heap** is modified to 40. Show the resulting **min-heap** after the modification.
min – heap property is **A[*PARENT(i)*] <= A[*i*]**

(5 marks)

Question 4 **(25 marks)**

a)

i. If modulo value is $q = 10$, how many spurious hits and valid hits do the **Rabin -Karp matcher** encounter in the text T = 6750100502007 when looking for pattern $P = 50$? (4 marks)

ii. How do you reduce the number of spurious hits in a) i)? (2 marks)

iii. What should be the number of spurious hits and valid hits if the best-case scenario occurs in Rabin-Karp algorithm? (2 marks)

b) Draw the state transition diagram for a string-matching automation for the pattern P = ***abaa*** and take the input alphabet as $\{a,b\}$

(6 marks)

c) Following is the **Naïve-String-Matcher** algorithm, which is used to find the occurrence(s) of a pattern string within another string or body of text.

Naïve-String-Matcher (T, P)

```
1.    n = T.length
2.    m = P.length
3.        for s = 0 to n-m
4.            if P[1..m] = T[s+1..s+m]
5.            then print "Pattern occurs with shift" s
```

Given the text and pattern as follows;

Text ***T***

a	b	a	b	a	a	a	a	b

Pattern ***P***

a	a	a

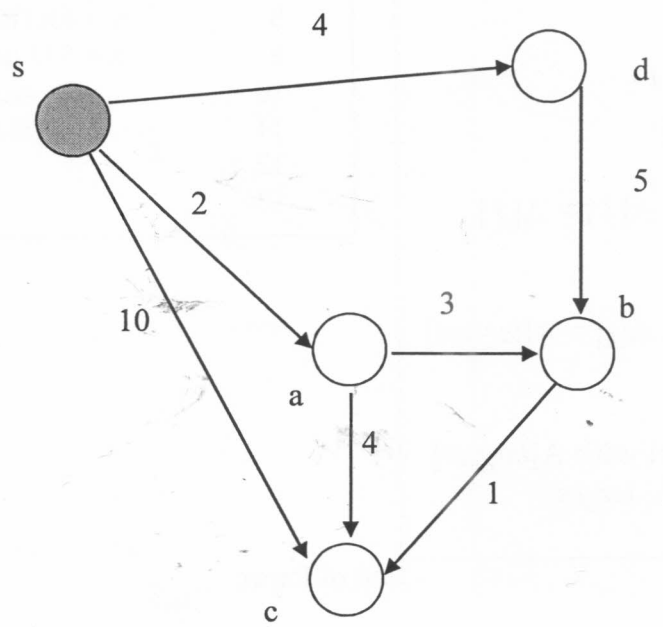
i) How many comparisons would occur in this algorithm? (1 mark)

ii) Show that the worst-case time complexity of the above algorithm is $O(m(n-m+1))$ where n is the number of characters in the text and m is the number of characters in the pattern.

(3 marks)

d) What is meant by the "**Minimum Cost Spanning Tree**" (MCST)? (2 marks)

e) Apply the **Dijkstra's** algorithm given at the end of the paper to find the shortest path from the source vertex ***s*** to all the other vertices of the graph. (For the purpose of illustration, assign the values only once to the given algorithm and use diagrammatic way to reach the answer.) (5 marks)



```
HEAP_INSERT(A, key)

1.      A.heap_size = A.heap_size +1
2.      i = A.heap_size
3.      while i > 1 and A[PARENT(i)] < key
4.              A[i] = A[PARENT(i)]
5.              i = PARENT(i)
6.      A[i] = key
```

```
HEAPSORT(A)
1.BUILD_HEAP[A]
2.for  i = A.length down to 2
3.    Exchange A[1] with A[i]
4.    A.heap_size = A.heap_size -1;
5.    HEAPIFY(A,1)


BUILD_HEAP (A)
1.  A.heap_size = A.length
2.  for i = ⌊A.length/2⌋ downto 1
3.          HEAPIFY(A,i)


HEAPIFY (A,i)
1.  l = LEFT_CHILD (i);
2.  r = RIGHT_CHILD (i);

3.  if l ≤ A.heap_size and A[ l ] > A[ i ]
4.          largest = l;
5.   else largest = i;
6.  if r ≤ A.heap_size and A[r] > A[largest]
7.          largest = r;
8.  if largest ≠ i
9.              exchange A[i] with A[largest]
10.             HEAPIFY (A, largest)
```

```
DIJKSTRA(G, w, s)

1   for each vertex v ∈ V[G]
2         d[v] = ∞
3         π[v] = NIL
4   d[s] = 0
5   S = Ø
6   Q = V[G]
7    while Q ≠ Ø
8           u = EXTRACT-MIN(Q)
9           S = S U {u}
10              for each vertex v ∈ Adj[u]
11                  if d[v] > d[u] + w(u, v)
12                      d[v] = d[u] + w(u, v)
13                      π[v] = u
```

End of Paper

Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination
Year 2, Semester 2 (2019)

IT2070 – Data Structures and Algorithms

Duration: 2 Hours

October 2019

Instructions to Candidates:

- This paper has 4 questions.
- Answer all questions in the booklet given.
- The total marks for the paper is 100.
- This paper contains 9 pages, including the cover page.
- Electronic devices capable of storing and retrieving text, including calculators and mobile phones are not allowed.
- This paper is preceded by 10 minutes reading period. The supervisor will indicate when answering may commence.

Question 1 **(25 marks)**

a) A Stack class called **StackX** has been created to store 5 integer values. push(), pop(), isEmpty(), isFull() and size() methods are available in the class.

i) Draw separate stack frames after performing the below operations to an empty stack. (4 marks)

```
s.push(5);
s.push(s.peek());
s.push(s.pop());
s.pop();
```

ii) Find the errors in the push method implemented below and correct them (3 marks)

```
public void  push(int j) {
        if (top == maxSize)
                system.out.println("Stack is full");
        else {
                stackArray[top] = j;
                top++;
        }
}
```

iii) Numbers are stored in a stack object called **s1**. Write the code segment written in main application to remove the odd numbers from the stack. Even numbers should remain in the stack. (5 marks)

b)

i) Draw the queue frames after performing the given code segment. A circular queue is used. (4 marks)

```
QueueX Q = new Queue(10);
int no = 10;
for(int i=1; i<=8; i++) {

   Q.insert(no);
   no = no + 10;
}
for(int i=1; i<=4; i++) {

      Q.insert(Q.remove());
}
```

ii) What is the result you get if you remove all the items from the above queue? (2 marks)

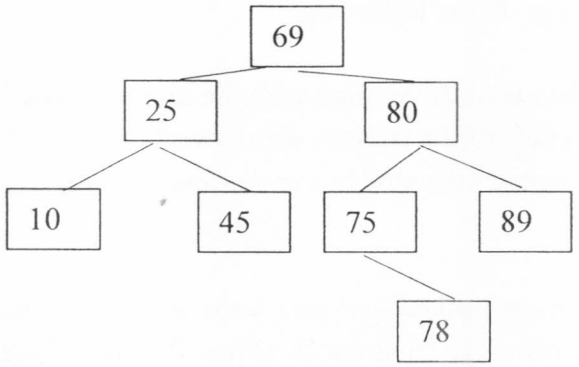
iii) getSize() method of a circular queue class returns the no of items in the queue. Implement getSize() method of the circular queue class. (2 marks)

c) Assume a queue class and stack class have been implemented. Using them, write code segment to reverse the given items in a queue. (i.e. After executing the code the items in the queue should be reversed)
Assume myStack and myQueue objects have already been created using the implemented classes and values are already available in the queue.

(5 marks)

Question 2 **(25 marks)**

a) Consider the following binary search tree and answer the questions



i) If the value 72 is inserted into this tree, which node becomes its parent? (1 mark)
ii) If the value 48 is inserted into this tree, which node becomes its parent? (1 mark)
iii) List all non leaf nodes in this tree. (1 mark)
iv) If node 25 is deleted which node should be its replacement node? (1 mark)
v) Find the depth of this tree (1 mark)
vi) Is this a complete binary tree (1 mark)

b) Consider the following classes to answer the below question. Assume the following classes have been implemented

Student class

int studentID int marks Student leftChild; Student rightChild;
void displayDetails();

Tree class

Student root
void insert(int marks, int studentID) Student minimum( ) Student maximum ( ) void descendingOrder ( )

Students of a class are stored in a tree data structure. Mark of the student is considered as the key to store them in the tree. Assume the tree object **studentMarks** is created with student details.

Write a code segment to do the following,

i) Display the details of the student with the highest mark (2 marks)
ii) Display the details of the student with lowest mark (2 marks)
iii) Display the marks of the students in descending order. (2 marks)

c) i) Consider the following LinkedLists and write a code segment that would change the LinkedList from Diagram A to Diagram B. (First, P1, and P2 are references to given links) (3 marks)

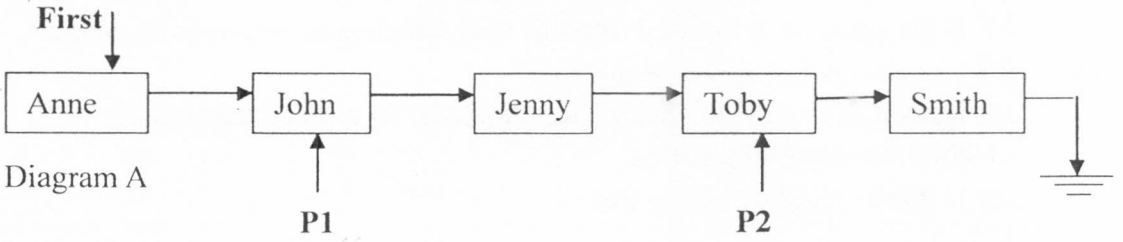


Diagram A

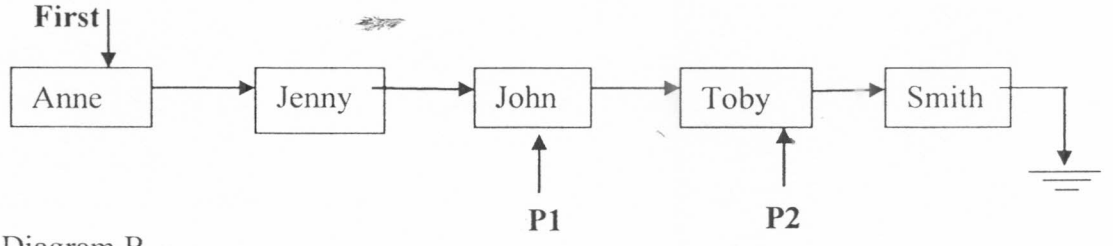
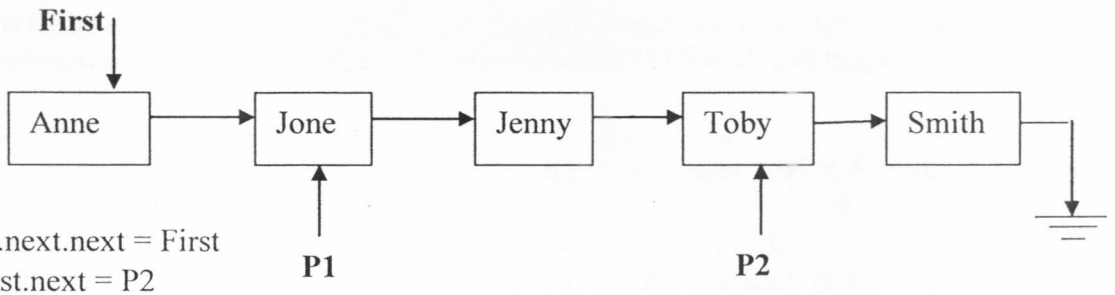


Diagram B

ii) Consider the following linked list. Illustrate how the linked list changes when each code segment is executed on the linked list. (3 marks)



P1.next.next = First
First.next = P2
First = P2.next
First.next = P1
P2.next = NULL

d) A library maintains the details of their books using a linked list. A book has a

Book No	integer
Number of copies	double

Assume the **book** class and the **linked list** class has already been implemented.

i) Implement a method called isAvailable() in the linked list class to return the reference of the link if the library has a given book. The book number is given as a parameter. (4 marks)

ii) Implement a method called lending( ) in the linked list class to update the number of copies (decrement by 1) when the book number is given as a parameter.
Hint : use the isAvailable( ) method to check the availability. (3 marks)

Question 3 **(25 marks)**

a) What is the running time of the below method using Big O notation? Explain your answer. (4 marks)

```
public int add100(int[] array) {

   if (array.length < 100)
      return 0;

   int sum = 0;
   for (int i = 0; i < 100; i++)
      sum += array[i];

   return sum;
}
```

b)

i) Following quicksort algorithm has errors. State the line numbers with errors and rewrite the algorithm. (PARTITION algorithm is given at the end of the question paper) (4 marks)

```
QUICKSORT (Arr, low, high)
1 if high < low
2    p = PARTITION(Arr, low, high)
3    QUICKSORT (Arr, low, p)
4    QUICKSORT (Arr, p, high)
```

ii) Describe the worst case of the of the quick sort algorithm using a diagram and find its recurrence equation. (2 marks)

iii) Solve the above equation using repeated substituted method and find the worst case running time in Theta notation. (4 marks)

c)

i) What is a max heap property? (1 mark)

ii) Consider the following array representation of a heap

A

60	50	45	30	20	35	15	10

Is this a max heap? Justify your answer. (3 marks)

d) Consider the below HEAP_EXTRACT_MAX and MAX_HEAPIFY algorithms used to return the maximum value from a max-heap.
Convert them to **HEAP_EXTRACT_MIN** algorithm which would remove and return the minimum value from a min heap. (7 marks)

```
HEAP_EXTRACT_MAX(A[1 .. n])
1. if A.heap_size >= 1
2.    max = A[1]
3.    A[1] = A[A.heap_size]
4.    A.heap_size = A.heap_size -1
5.    MAX_HEAPIFY(A,1)
6.    return max
```

```
MAX_HEAPIFY (A,i)
1.  l = LEFT(i);
2.  r = RIGHT(i);
3.  if l ≤ A.heap_size and A[ l ] > A[ i ]
4.          largest = l;
5.  else largest = i;
6.  if r ≤ A.heap_size and A[r] > A[largest]
7.          largest = r;
8.  if largest ≠ i
9.          exchange A[i] with A[largest]
10.         MAX_HEAPIFY (A,largest)
```

Question 4 **(25 marks)**

a) Following is the **Naïve-String-Matcher** algorithm, which is used to find the occurrence(s) of a pattern string within another string or body of text.

```
Naïve-String-Matcher (T, P)
1.    n = T.length
2.    m = P.length
3.        for s = 0 to n-m
4.            if P[1..m] = T[s+1..s+m]
5.            then print "Pattern occurs with shift" s
```

Given the text and pattern as follows;

Text ***T***

a	b	b	a	c	d	a	a	b

Pattern ***P***

a	a	a

i) How many comparisons would occur in this algorithm? (2 mark)

ii) How many valid and invalid shifts would occur in this algorithm? (2 marks)

b)

i. If modulo value is $q = 100$, how many spurious hits and valid hits do the **Rabin -Karp matcher** encounter in the text T = 203410052006 when looking for pattern $P = 100$? (4 marks)

ii. How do you reduce the number of spurious hits in b) i)? (2 marks)

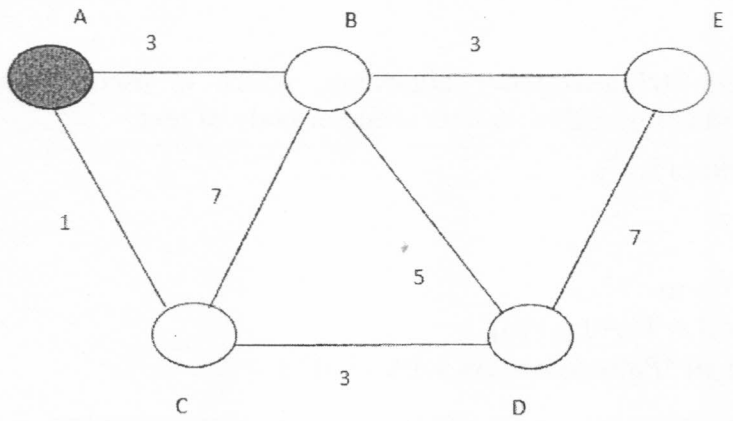
iii. What should be situation of the worst-case scenario occurs in Rabin-Karp algorithm? (2 marks)

c) Draw the state transition diagram for a string-matching automation for the pattern P = ***aba*** and take the input alphabet as $\{a, b, c\}$

(7 marks)

d) What is meant by the "**Optimal Solution**" in Greedy method? (1 mark)

e) Apply the **Dijkstra's** algorithm given at the end of the paper to find the shortest path from the source vertex ***s*** to all the other vertices of the graph. (For the purpose of illustration, assign the values only once to the given algorithm and use diagrammatic way to reach the answer.) (5 marks)




```
DIJKSTRA(G, w, s)
1   for each vertex v ∈ V[G]
2         d[v] = ∞
3         π[v] = NIL
4   d[s] = 0
5   S = Ø
6   Q = V[G]
7   while Q ≠ Ø
8         u = EXTRACT-MIN(Q)
9         S = S U {u}
10            for each vertex v ∈ Adj[u]
11                if d[v] > d[u] + w(u, v)
12                    d[v] = d[u] + w(u, v)
13                    π[v] = u
```

```
PARTITION(A, p, r)
1   x = A[r]
2   i = p - 1
3   for j = p to r - 1
4       if A[j] ≤ x
5           i = i + 1
6           exchange A[i] with A[j]
7    exchange A[i + 1] with A[r]
8    return i + 1
```

End of Question Paper

00284

Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination

Year 2, Semester 2 (2022)

IT2070 – Data Structures and Algorithms

Duration: 2 Hours

June 2022

Instructions to Candidates:

- This paper has 4 questions.
- Answer all questions.
- The total marks for the paper is 80.
- Electronic devices capable of storing and retrieving text, including calculators and mobile phones are not allowed.

Question 1 **(20 marks)**

1) Taking modulo $q = 100$, how many spurious hits and valid hits do the Rabin -Karp matcher encounter in the text T = 900800600300900 when looking for pattern $P = 600$? (2 marks)

2) Draw the state transition diagram for a string-matching automation for the pattern P = ***abab*** and take the input alphabet Σ as $\{a,b\}$. (6 marks)

3) Following is the **Naïve-String-Matcher** Algorithm, which is used to find the occurrence(s) of a pattern string within another string or body of text. (4 marks)

Naïve-String-Matcher (T, P)

1. n ← length[T]
2. m ← length[P]
3. for s ← 0 to n-m
4. do if P[1..m] = T[s+1..s+m]

Given the text and pattern as follows; (2 marks)

Text ***T***

1	0	0	0	1	0	0	0	1

Pattern ***P***

0	0	0

a) How many comparisons would occur in this algorithm?

b) Show that best-case time complexity of the above algorithm is $O((n-m+1))$ where n is the number of characters in the text and m is the number of characters in the pattern.

4) Given a chain $(A_1, A_2, ..., A_{n-1}, A_n)$ of n matrices, where for i = 1,2, …, n matrix A_i has dimension $p_{i-1} \times p_i$. Assume that $m[i,j]$ is the minimum number of scalar multiplications needed to compute the matrix $A_{i...j} = A_i \times A_{i+1} \times ..... \times A_{j-1} \times A_j$ and it is defined below

$$m[i,j] = \begin{cases} 0 & \text{if } i = j \\ \min\limits_{i \le k < j} \{m[i,k] + m[k+1,j]\} + p_{i-1} p_k p_j & \text{otherwise} \end{cases}$$

Consider the following set of metrics A_1, A_2, A_3 and A_4 with their dimensions of 2×5, 5×5, 5×10 and 10×20 respectively. (8 marks)

a) Draw the m and s table to find the optimal parenthesizing of the matrices for the above sequence of matrices using the dynamic programming algorithm.

b) Hence find the optimal parenthesizing and optimal number of scalar multiplications of the above matrices

Question 2 **(20 marks)**

1) The power function can be defined as $pow(x, n) = x^n$. This can be evaluated using the multiplication as $x^n = x \times x^{n-1}$ where x is any real number and n is a non-negative integer. [Hint: $pow(x, n-1) = x^{n-1}$] (8 marks)

   a) Write a recursive relation for $pow(x, n)$ where x is any real number and n is a non-negative
   b) Write a recursive algorithm in pseudo code for the above recursive relation.
   c) Write a recurrence equation that describe the running time T (n) for the above part b) recursive algorithm.

2) Analyze the running time of the following program fragment assuming a **RAM model** of computation. (4 marks)

```
sum ← 0
for i ← n down to 0
    sum = sum + 1
```

3) Given below is an algorithm for *QUICKSORT* (*A, p, r*)

```
Procedure QUICKSORT (A, p, r)
1  if p < r
2    then q ← PARTITION(A, p, r)
3         QUICKSORT (A, p, q-1)
4         QUICKSORT (A, q+1, r)
```

```
Procedure PARTITION(A, p, r)
1  x ← A[r]
2  i ← p - 1
3    for j ← p to r - 1
4      do if A[j] ≤ x
5           then i ← i + 1
6                exchange A[i] ↔ A[j]
7    exchange A[i + 1] ↔ A[r]
8  return i + 1
```

Illustrate the operations of the *QUICKSORT (A, p, r)* for the array with the given set of elements. (For the illustration process assign the values only once to the given algorithm codes and then use diagrammatic way to reach the answer.) (4 marks)

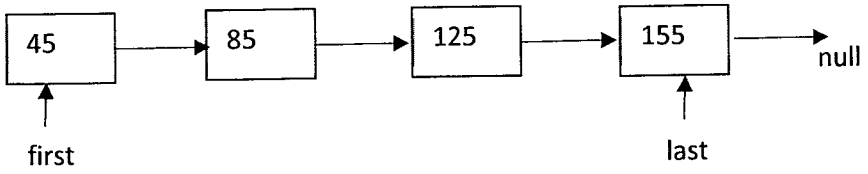
1	2	3	4	5	6
28	15	1	30	0	8

4) Is the sequence representing <40, 25, 19, 6, 13, 18, 20, 5, 7, 12> a max heap? Justify your answer. (4 marks)

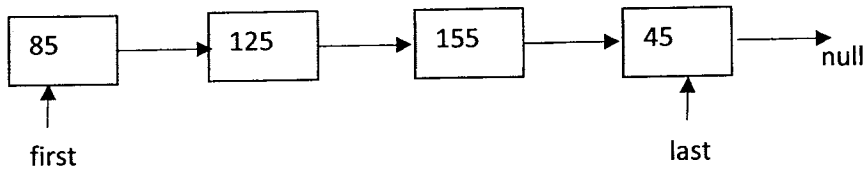
Question 3 **(20 marks)**

1) Consider the following two link lists. Write the code segment to convert link list A to link list B. (4 marks)

Link list A



Link list B



2) Consider the below link class and implement the method deleteLast() in the link list class to remove the last link from the linklist. (4 marks)

Link class

int iData; Link next;
Link(int id) void displayLink()

3) Following numbers are inserted to a binary search tree.

45, 67, 12, 6, 20, 80, 50

i. Draw the binary search tree after inserting all the numbers (3 marks)

ii. What type of a binary tree do you get in i) above? (1 mark)

iii. What is the height of the tree in i) above? (1 mark)

iv. Assume you have N nodes in a binary search tree of the type given in i) above. Find the running time in Big O notation when searching a value from this tree. Give reasons for your answer. (3 marks)

v. What is the running time in Big O notation when searching a value from a link list with N no of links? Give reasons for your answer. (2 marks)

vi. Comment on the speed of the two data structures (with large N values) when searching a value. (Hint: use the answers given in iv) and v) above) (2 marks)

Question 4 **(20 marks)**

1) State one similarity and one difference between Stack and Queue data structure. (2 marks)

2) You are expected to develop a function to delete the middle element of a given stack. Note that the stack class is already implemented. You may assume that the stack is implemented to store integers. (9 marks)

Hint: Use two stacks (stack objects are '"s" and "s1")

Sample input and output is given below.

Input: `Stack[] = [1, 2, 3, 4, 5]`

Output: `Stack[] = [1, 2, 4, 5]`

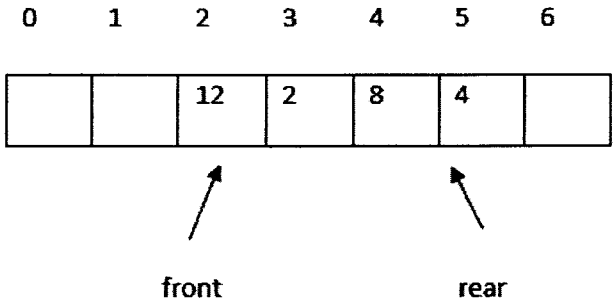
Input: `Stack[] = [1, 2, 3, 4, 5, 6]`

Output: `Stack[] = [1, 2, 3, 5, 6]`

The function signature is as follows.

```
public void deleteMiddle()
{   }
```

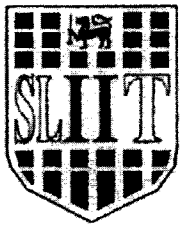
3) (c) Consider the following circular queue which is implemented to store integers. Draw the queue frames after executing each of the statements. (5 marks)



(i) insert (10)
(ii) peekFront()
(iii) insert (22)
(iv) remove ()
(v) insert (18)

5) Consider a linear queue to store double values. Implement a method to calculate the mean value of the currently available values in the queue and insert it to the same queue. Note that the queue class is already implemented and assume the queue object is "q". (4 marks)

00716



Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination
Year 2, Semester 2 (2022)

IT2070 – Data Structures and Algorithms

Duration: 2 Hours

November 2022

Instructions to Candidates:

- This paper has 4 questions.
- Answer all questions in the booklet given.
- The total marks for the paper is 100.
- This paper contains 8 pages, including the cover page.
- Electronic devices capable of storing and retrieving text, including calculators and mobile phones are not allowed.
- 10 minutes reading time is allowed

Question 1 **(25 marks)**

Consider the StackX class shown below

int[] stackArray int maxSize int top
void push(double j) double pop( ) double peek( ) boolean isEmpty( ) boolean isFull( )

a) An object of StackX class is created as myStack with size 5. Start from the stack frame given in Figure 1 and draw the stack frame after each statement given below. (5 marks)

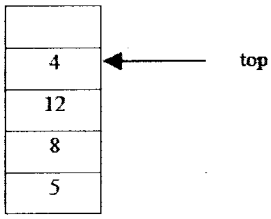


Figure 1

i) mySatck.push(6);
ii) myStack.peek();
iii) myStack.push(2);
iv) myStack.pop( );
v) myStack.pop( );

b) Assuming that the StackX class has been implemented using java, complete the following java program **StackApp** to do the following.

```
class StackApp {
     public static void main(String[] args) {
     ------------------------------------
     ------------------------------------
     ------------------------------------
}
```

i) Create a stack object called **st** of size seven. (2 marks)

ii) Store the integers given below.

20, 43, 12, 37 (2 marks)

iii) Read another integer from the keyboard. (1 mark)

iv) Find out whether the number entered in iii) exists in the stack. (Hint : can use another stack object if necessary) (5 marks)

c) Give your opinion about the following (4 marks)

i) The array implementation of stack and queue has some limitations.

ii) Circular queue allows to insert and remove items from the rear.

d) Consider the following QueueX class

```
class QueueX {
      private int maxSize;
      private int [] queArray;
      private int front;
      private int rear;
      private int nItems;
  ..........
  ..........
  }
```

Complete the following table with correct boolean expression used in the given method. (6 marks)

	Linear Queue	Circular Queue
boolean isEmply( )		
boolean isFull( )		

Question 2 **(25 marks)**

a) Draw the link list created after executing the below code segment. insertFirst() method will insert the link as the first link and insertLast() method inserts the link as the last link. deleteFirst() method removes the first link from the link list. (3 marks)

```
linkList myList = new linkList();
myList.insertFirst(100);
myList.insertFirst(50);
myList.insertFirst(10);
myList.insertLast(200);
myList.deleteFirst();
```

b) In a hospital management system, patient beds are stored in a link list. patientBed class and linkList classes are given below.
'Vacant' attribute of patientBed class will be 0 when the patient is admitted to the hospital and assigned the bed. It will be 1 when the patient is discharged.

Assume the two classes have already been implemented.

patientBed

int bedNo int vacant patientBed next
patientBed (int bedNum) void displayDetails() void assignBed() void discharge()

linkList

patientBed first;
void linkList() boolean isEmpty() void addFirst(int bedNo) patientBed findVacant()

i) You need to modify the linkList class by adding another method called countVacantBeds() in the linkList class. This method returns the number of vacant patient beds. Implement the method. (5 marks)

ii) findVacant() method in the linkList class returns the first vacant patientBed. If there are no vacant beds, it will return NULL. Write code segment in your main application to find a vacant bed and assign the bed. Display the assigned bedNo. If there are no vacant beds in the hospital display the message "Patient beds are not available". (5 marks)

c)

i) Insert the following nodes to a binary search tree (3 marks)

88, 99, 77, 80, 90, 60

ii) Find the height of the above tree (1 mark)

iii) Display the nodes of the above tree using post order traverse method (3 marks)

iv) Assume that the below methods are available in tree class to traverse the tree.

```
private void inOrder(Node localNode)
private void preOrder(Node localNode)
private void postOrder(Node localNode)
```

Implement a method of the tree class called displayGreaterThanRoot() to display all the nodes greater than the root in ascending order. (5 marks)

Question 3 **(25 marks)**

a) Find the step count of the following pseudocode using RAM Model. (7 marks)

i)

```
int j = 0
for (i = 1 to 5)
     print i
     while (j < 5)
          j = j + 1
          print j
```

ii)

```
int i = 1
while (i <= 0)
     i = i + 2
     print i
```

b) State the complexity using Big(O) notations of the following running time equations. (6 marks)

i) $T(n) = n + n \log n + n^2 + 5$

ii) $T(n) = 2n + 10$

iii) $T(n) = 3^n + n! + 4n^2 + n^n - 8$

c)

i) What is the recurrence equation for the Merge Sort Algorithm? (2 marks)

ii) State the reason of having only the best case in the Merge Sort Algorithm. (2 marks)

Pseudo code of the MERGESORT algorithm is given below

```
MERGESORT (A, p, r)
1.    if p < r
2.       q = ⌊(p+r )/2⌋
3.       MERGESORT (A, p, q)
4.       MERGESORT (A, q+1, r)
5.       MERGE (A, p, q, r)
```

d) Illustrate the operations of **PARTITION(A, p, r)** in Quicksort Algorithm for array A={8,2,10,3,5} (8 marks)

Pseudo code of the PARTITION algorithm is given below

```
PARTITION(A, p, r)
1      x = A[r]
2      i = p - 1
3      for j = p to r - 1
4          if A[j] ≤ x
5              i = i + 1
6              exchange A[i] with A[j]
7      exchange A[i + 1] with A[r]
8      return i + 1
```

Question 4 **(25 marks)**

a)

i) Pseudo code of the BUILD_MAX_HEAP algorithm is given below. Complete line number 2 of the algorithm and explain the purpose of that line. (4 marks)

```
BUILD_MAX_HEAP (A)
1.   A.heap_size =A.length
2.   for i =  ..... downto 1
3.      MAX_HEAPIFY(A,i)
```

ii) Identify the violating node of the below max heap and illustrate the operations of the MAX_HEAPIFY (A,i) for the array A given below. (Use diagrammatic way to reach the answer) (6 marks)

Array A

10	90	60	50	20	35	40	15

Pseudocode of MAX_HEAPIFY is given below

```
MAX_HEAPIFY (A,i)
1.    l =  LEFT_CHILD (i);
2.    r =  RIGHT_CHILD (i);
3.    if l ≤ heap_size[A] and A[ l ] > A[ i ]
4.         largest = l;
5.    else largest = i;
6.    if r ≤ heap_size[A] and A[r] > A[largest]
7.         largest =  r;
8.    if largest ≠  i
9.         exchange A[i] ↔ A[largest]
10.        MAX_HEAPIFY (A, largest)
```

b) Answer the following question based on the Rabin – Karp algorithm
Taking modulo q = 10, how many spurious hits and valid hits would encounter in the text T = 902883280088 when looking for pattern p = 28? (5 marks)

c) Draw the state transition diagram for a string-matching automation for the pattern P = abb and take the input alphabet as {a,b} (6 marks)

d) Following is the Naïve-String-Matcher Algorithm, which is used to find the occurrence(s) of a pattern string within another string or body of text.

```
Naïve-String-Matcher (T, P)
1. n =  length[T]
2. m =  length[P]
3. for s =  0 to n-m
4.    if P[1..m] = T[s+1..s+m]
5.       print "Pattern occurs with shift" s
```

Consider the below text

a	A	b	b	b	a	a	a	b	a

How many comparisons would occur in this algorithm for the pattern "abb"? (4 marks)

End of Question Paper

00439

Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination (June Intake)
Year 2, Semester 2 (2023)

IT2070 – Data Structures and Algorithms

Duration: 2 Hours

June 2023

Instructions to Candidates:

- This paper has 4 questions.
- Answer all questions in the booklet given.
- The total marks for the paper is 100.
- This paper contains 9 pages, including the cover page.
- Electronic devices capable of storing and retrieving text, including calculators and mobile phones are not allowed.
- 10 minutes reading time is allowed

Question 1 **(25 marks)**

a) What are the basic features in the following data structures in relation to the basic operations insert, delete and retrieval data. (3 marks)

	Insert	Delete	Retrieve
Stack			
Linear Queue			
Circular Queue			

b) Fill in the following blanks. (6 marks)

i) Most popular data structure is _____.

ii) The process of retrieving the element at the top of a stack without removing it is called_____

iii) A stack can be implemented using _____ or _____

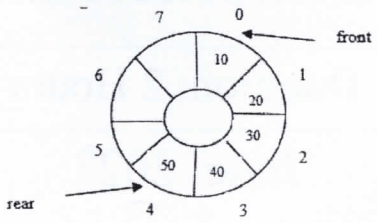
iv) In a stack, if the "top" pointer is equal -1, then the stack is _____

v) The time complexity of the push and pop operations on a stack implementation are; push -_____ and pop - _____

c) Consider the following circular queue with initial values given below.

Initials values are :

Front=0
Rear=4
Count=5



Show how the above parameter values (Front, Rear and Count) will change after each of the operations by completing the table given below. (6 marks)

i) insert(60)
ii) insert(70)
iii) delete( )
iv) insert(80)
v) insert(90)
vi) insert(100)

Operation	**Front**	**Rear**	**Count**
insert(60)			
insert(70)			
----------			

d) Consider the UML class diagrams given below.

StackX
- double[] stackArray - int maxSize - int top
+void push(double j) +double pop( ) +double peek( ) +boolean isEmpty( ) +boolean isFull( )

QueueX
- StackX s1 - StackX s2
+ void insert(double j) + double remove( ) + double peekFront( ) + boolean isEmplty( ) + boolean isFull ( )

A queue data structure can be implemented using two stacks. Assume the size of these stacks as 5. Assume that the java implementation of the StackX class is given.

Complete the following java code for the QueueX class. (6 marks)

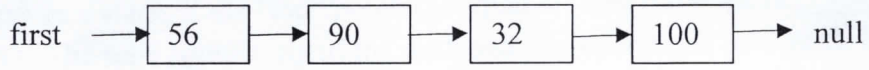
```
public class QueueX
{
     private StackX s1;
     private StackX s2;

     public QueueX(int x)
     {
          ..........//Line 1
          ..........//Line 2
     }

     public void insert(double j)
     {
          ..........//Line 3 - insert an element to stack 1
     }

     public double remove( )
     {
          if( s2.isEmpty( )){
               while(!s1.isEmpty()){
                    s2.push( s1.pop())
               }

          }
          ..........//Line 4
     }
```

e) Consider the Queue implementation given in part d) above, draw the status of the two stacks, after the execution of the following code. (4 marks)

```
class QueueApp {
     public static void main(String[] args) {
          QueueX q = new QueueX(5);
          q.insert(5.0);
          q.insert(3.0);
          double temp = q.remove( );
          q.insert(10.0);
     }
}
```

Question 2 **(25 marks)**

a) Consider the below link list (4 marks)



Display the output and draw the link list after executing the following code segments

```
first = first.next;
System.out.print(first.next.ID);
first.next = NULL;
```

b) Consider the below Link class and the LinkList class. Assume the classes have already been implemented. addFirst() method in the LinkList class inserts a new link as the first link and removeFirst() method deletes the first link of the link list.

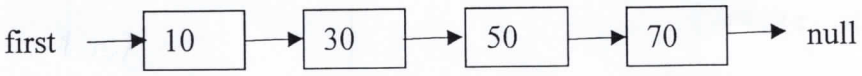
Link
+int ID +Link next
+ Link (int IDNo)

LinkList
-Link first
+void LinkList() +boolean isEmpty() +void addFirst(int ID) +int removeFirst()

i) Modify the LinkList class by adding another method called deleteAllLinks(). This method deletes all the links in the link list and display the IDs of the deleted links. Implement the method deleteAllLinks(). (5 marks)

ii)

1) Consider the below link list. What is the output you get after calling deleteAllLinks() method? (2 marks)



2) Draw the above link list after executing deleteAllLinks() method. (2 marks)

iii) Modify the method implemented in b) i) to remove all the links except the last link. (3 marks)

c) Key values of a binary search tree is given below.

90, 34, 100, 23, 50, 98, 125, 10

i) Insert the above values in a binary search tree. (2 marks)
ii) What type of a binary tree did you get in c) i)? (1 mark)
iii) How do you convert the above tree to a full binary tree? (1 mark)
iv) Analyze and discuss the importance of a full binary tree when searching a value from a tree data structure (5 marks)

Question 3 **(25 marks)**

a) Consider the following pseudocode and find the complexity using Big O notation. (3 marks)

```
for (i = 1 to 5)
     print i
for (j = 1 to n)
     print j
```

b) PARTITION() algorithm which is used in Quick Sort algorithm is given below. It returns the partition index.

i) Consider the following array A and illustrate the steps of applying PARTITION() algorithm to array A (6 marks)

Array A

1	2	3	4	5
5	4	0	8	6

```
PARTITION(A, p, r)
1   x = A[r]
2   i = p - 1
3   for j = p to r - 1
4     if A[j] ≤ x
5           i = i + 1
6           exchange A[i] with A[j]
7   exchange A[i + 1] with A[r]
8   return i + 1
```

ii) Which value has been selected as the pivot element? (2 marks)

iii) What is the partition index computed by the partition algorithm? (3 marks)

c)

i) What is the best case in Quick Sort algorithm? (2 marks)

ii) Obtain the recurrence equation for the best case in Quick Sort algorithm? (2 marks)

iii) Master Theorem can be applied to recurrences of the form $T(n) = a\,T(n/b) + f(n)$. Solve the recurrence equation obtained in c) ii) using Master Theorem and find the complexity. (3 marks)

Master Theorem

$$T(n) = \begin{cases} \Theta\left(n^{\log_b a}\right) & f(n) = O\left(n^{\log_b a - \varepsilon}\right) \rightarrow f(n) < n^{\log_b a} \\ \Theta\left(n^{\log_b a} \lg n\right) & f(n) = \Theta\left(n^{\log_b a}\right) \rightarrow f(n) = n^{\log_b a} \\ \Theta\left(f(n)\right) & f(n) = \Omega\left(n^{\log_b a + \varepsilon}\right) \rightarrow f(n) > n^{\log_b a} \\ & \text{if } af(n/b) \le cf(n) \text{ for c} < 1 \text{ and large } n \end{cases}$$

```
QUICKSORT (A,p,r)
 1 if p < r
 2    q = PARTITION(A, p, r)
 3    QUICKSORT (A, p, q-1)
 4    QUICKSORT (A, q+1, r)
```

d) Figure (1) is the pseudocode of the Insertion Sort algorithm. Describe the impact of the output and the performance if line (7) of the same pseudocode is changed as shown in Figure (2). (Line 7 in Figure(2) is inside the while loop) (4 marks)

```
INSERTION-SORT(A)
1 for j = 2 to A.length
2     key = A[j]
3     i = j - 1
4     While i > 0 and A[i] > key
5       A[i+1] = A[i]
6       i = i-1
7     A[i+1]  = key
```

Figure (1)

```
INSERTION-SORT(A)
1 for j = 2 to A.length
2     key = A[j]
3     i = j - 1
4     While i > 0 and A[i] > key
5       A[i+1] = A[i]
6       i = i-1
7       A[i+1]  = key
```

Figure (2)

Question 4 **(25 marks)**

a) What is a max heap? Explain your answer using an example. (3 marks)

b) Identify the violating node of the below max heap and illustrate the operations of the MAX_HEAPIFY (A,i) for the array A given below. (Use diagrammatic way to reach the answer) (3 marks)

Array A

60	40	58	10	44	55	33	2	6	42	15

Pseudocode of MAX_HEAPIFY is given below

```
MAX_HEAPIFY (A,i)
1.    l =  LEFT_CHILD (i);
2.    r =  RIGHT_CHILD (i);
3.    if l ≤ heap_size[A] and A[ l ] > A[ i ]
4.          largest = l;
5.    else largest = i;
6.    if r ≤ heap_size[A] and A[r] > A[largest]
7.          largest =  r;
8.    if largest ≠  i
9.          exchange A[i] ↔ A[largest]
10.         MAX_HEAPIFY (A, largest)
```

c) Consider the HEAPSORT algorithm is given below.

```
HEAPSORT(A)
1.    BUILD_MAX_HEAP[A]
2.    for  k = A.length down to 2
3.         exchange A[1] with A[k]
4.         A.heap_size = A.heap_size - 1;
5.         MAX_HEAPIFY(....., .....)
```

i) Complete line number 5 of the HEAPSORT algorithm. (2 marks)

ii) Explain the reason for the calling **MAX_HEAPIFY** algorithm in line number 5 with the arguments given in c) i) above. (4 marks)

iii) A student has written the above heapsort algorithm by changing line number 2 as follows,

```
for  k = A.length down to 1
```

Discuss how this modification affects the results and the performance (5 marks)

d)

i) Draw the state transition diagram for a string-matching automation for the pattern P = abba and take the input alphabet as {a,b} (6 marks)

ii) Consider the below text

a	b	b	a	b	b	a	a	b	a

Illustrate how you would find the pattern "abba" in the above text using your answer in part d) i) above. (2 marks)

End of Question Paper

00416

Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination
Year 2, Semester 2 (2023)

IT2070 – Data Structures and Algorithms

Duration: 2 Hours

November 2023

Instructions to Candidates:

- This paper has 4 questions.
- Answer all questions in the booklet given.
- The total marks for the paper is 100.
- This paper contains 8 pages, including the cover page.
- Electronic devices capable of storing and retrieving text, including calculators and mobile phones are not allowed.
- 10 minutes reading time is allowed

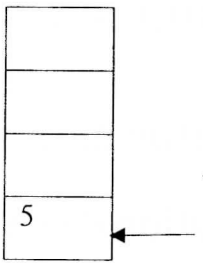
Question 1 (25 marks)

a) Write one practical example for each of the following (2 marks)

i) stack undo feature in text editor, page visit in browser history

ii) queue pipeline, printer queue

b) Consider the initial stack frame of a stack given below.

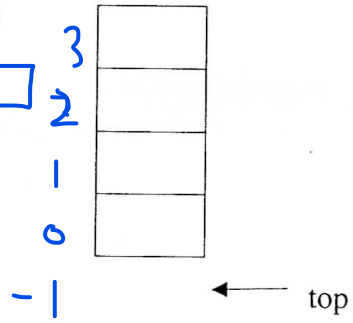


push()
pop()
peek()

Below stack frames are obtained after executing three operations one after another to the above stack frame. Write down the operations. (3 marks)

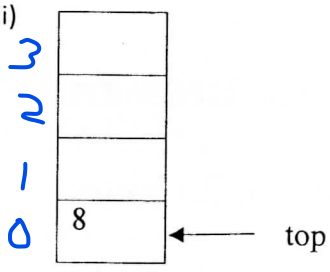
stack.pop();

i)



stack.push(8);

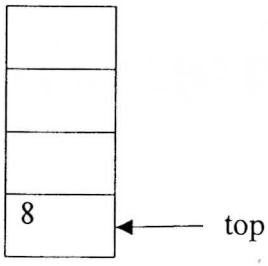
ii)

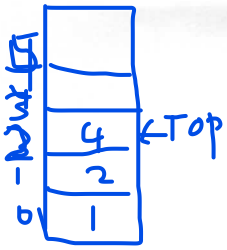


top + 1
0 + 1 = 1
nItems = top + 1
nItems = 0 + 1 = 1

stack.peek();

iii)





nItems = top + 1
= 2 + 1 = 3

c) Fill in the following blanks. (6 marks)

i) The "top" is -1 when the stack is empty.

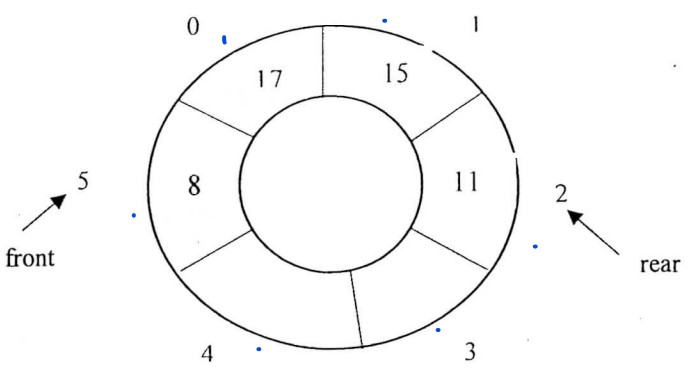
ii) The no of items in a stack can be obtained from (top + 1) .

iii) A stack can be implemented using Array or Linked List .

iv) "PeekFront" operation of a linear queue and circular queue are same (same/ different)

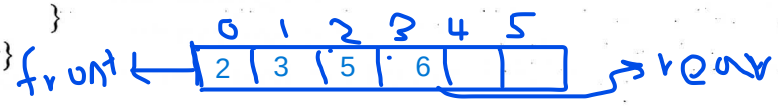
v) The time complexity of the "insert" operation on a queue implementation is o(n) .

d) Consider the below circular queue. A value is removed from the queue using the "remove" method given below. Write down the values of front after executing line 4, line 5, line 6 and line 8. (4 marks)



maxSize = 6

```
public int remove() {
1.      if ( nItems == 0)
2.          system.out.println("Queue is empty");
3.      else {                                  front = 5
4.          int temp = queArray[front++];  front = front + 1 = 5 + 1 = 6
5.          if (front == maxSize)true
6.              front  = 0;    front = 0
7.          nItems--;    front = 0
8.          return temp;    front = 0
9.      }
}
```



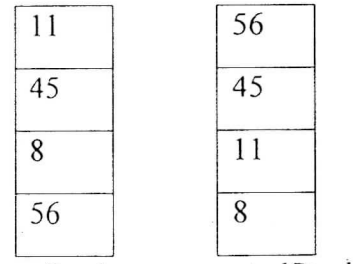
nItems = front - rear + 1
= 0 - 3 + 1
= -2

nItems = rear - front + 1
= 3 - 0 + 1
= 4

e) "front - rear + 1 can be used to find the no of items in a linear queue"
Do you agree with the above statement? If you agree, justify your answer using a diagram. If you do not agree, write the correct answer. (3 marks)

f) Write a code segment in your main program using java to sort the values in a stack in ascending order. Assume the stack object **myStack** is available with values and another empty stack object called **sortedStack** is already created. The sorted values should be stored in sortedStack. (7 marks)

Ex:




```
while (!myStack.isEmpty()) {
  double temp = myStack.Pop();

  while (!sortedStack.isEmpty() &&
sortedStack.Peek() > temp) {

myStack.Push(sortedStack.Pop());
  }

  sortedStack.Push(temp);
}
```

Consider the StackX class is given below.

StackX
- int[] stackArray - int maxSize - int top
+void push(double j) +double pop( ) +double peek( ) +boolean isEmpty( ) +boolean isFull( )

Question 2 **(25 marks)**

a) Consider the link list given in Figure (a)

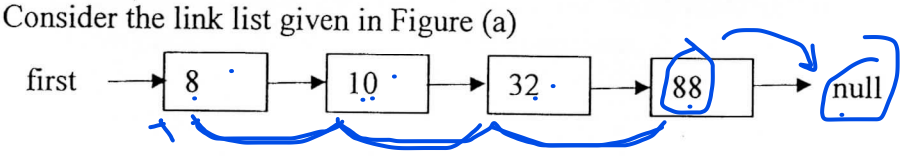


Figure (a)

Execute below code segments on the link list given in Figure (a). Display the output after executing the code segment (6 marks)

i)
```
Link temp = first;
while (temp != NULL)
{
   System.out.print(temp.ID);
   temp = temp.next;
}
```

8, 10, 32, 88

```
Link temp = first
while (temp != null){
   if (temp.next == null){
   S.O.P("Last link = " + temp);
  }
 temp = temp.next;
}
```

ii)
```
Link temp = first;
while (temp.next != NULL)
{
   System.out.print(temp.ID);
   temp = temp.next;
}
```

8,10,32

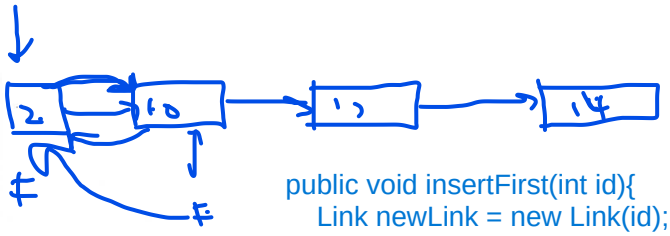
iii)
```
first = first.next;
System.out.print(first.next.ID);
```

32

b) Modify the code segment given in a) i) above to display the last link in a link list (4 marks)

c) Consider the following Link class of a **doubly link list**. Implement the insertFirst() method of the doubly link list class (4 marks)

Link
+int ID +Link next +Link previous
+ Link (int IDNo)




```
public void insertFirst(int id){
    Link newLink = new Link(id);

    newLink.next = first;
    first = newLink;
}
```

d) Key values of a binary search tree is given below.

50 65 25 30 10 58 70

i) Insert the above values in a binary search tree. (2 marks)

ii) What type of a binary tree did you get in d) i)? full binary tree (1 mark)

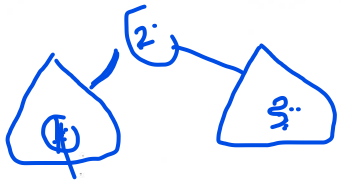
iii) Assume the values given above are sorted in ascending order first and then inserted to a binary search tree. Compare the tree you get with d) i) and discuss the performance. (3 marks)

e) Complete the method displayLeafNodes() of the tree class given below. This method will print all the leaf nodes in the tree.
Assume the Node class given below has been implemented. (5 marks)

Node
+int ID +Node leftChild +Node rightChild
+Node(int ID) +displayNode()

```
private void dislayLeafNodes(Node localNode){

    if (localNode != null){
        localNode.displayNode();
        displayLeafNode(localNode.leftChild);
        displayLeafNode(localNode.rightChild);
     }


}
```

```
private void  displayLeafNodes(Node localNode)
{

}
```



Question 3 (25 marks)

a) Find the step count of the following pseudocodes using RAM model. (6 marks)

i)

```
a = 5
while (a < 10)
    a = a + 2
    print a
```

ii)

```
i = 10
for ( j = 1 to 5 )
    while ( i >= 0 )
        i = i – 1
        print i
```

b) State the asymptotic notations in Big(O) notations for the above pseudocodes a) i) and a) ii). (4 marks)

c) PARTITION() algorithm which is used in Quick Sort algorithm is given below. It returns the partition index.

i) Consider the following array A and illustrate the steps of applying PARTITION() algorithm to array A (7 marks)

Array A

1	2	3	4	5
8	5	7	6	2

```
Procedure PARTITION(A,p,r)
1   x ← A[p]
2   i ← p -1
3   j ← r +1
4   while TRUE
5       do repeat j ← j - 1
6            until A[j] ≤ x
7          repeat i ← i + 1
8            until A[i] ≥ x
9          if i < j
10           then exchange A[i] ↔ A[j]
11           else return j
```

ii) If the values are in ascending order, comment on the performance of Quick Sort algorithm using the above Partition() algorithm. (3 marks)

iii) For the situation derived from ii), obtain the recurrence equation that describes the running time T(n). (2 marks)

iv) Solve the recurrence equation obtained in iii) using Repeated Substitution method and express the time complexity using Big(O) notation. (3 marks)

```
QUICKSORT (A,p,r)
 1 if p < r
 2    q = PARTITION(A, p, r)
 3    QUICKSORT (A, p, q-1)
 4    QUICKSORT (A, q+1, r)
```

Question 4 **(25 marks)**

a) State one difference between a queue and a priority queue. (1 mark)

b) If the following pseudocode is applied to the max heap A, what will be returned as the output? (1 mark)

A

1	2	3	4	5
70	60	30	20	10

```
HEAP_EXTRACT_MAX (A[1 .. n])
1. if A.heap_size >= 1
2.      max = A[1]
3.     A[1] = A[A.heap_size]

4.      A.heap_size = A.heap_size -1
5.     MAX_HEAPIFY(A,1)
6.     return max
```

c) Based on the pseudocode given in b) above, answer the following.

i) What is the purpose of executing line (4)? (2 marks)

ii) What is the purpose of executing line (5)? (2 marks)

d) Illustrate the steps to be followed in heap_insert() given below to the max heap A. Assume that a node with the value of 145 to be inserted as a new node. (7 marks)

A

1	2	3	4	5	6	7
200	185	100	50	40	80	90

HEAP_INSERT(A, key)
1. A.heap_size = A.heap_size + 1
2. i = A.heap_size
3. while i > 1 and A[PARENT(i)] < key
4. A[i] = A[PARENT(i)]
5. i = PARENT(i)
6. A[i] = key

e) i) Compute the number of valid shifts and invalid shifts occurred with Naïve String-Matching algorithm for following Text(T) and the Pattern(P). (2 marks)

T = abacbadab
P = cab

ii) Briefly explain the time complexity using Big (O) notation for the situation occurred in e) i) above. (2 marks)

f) i) Draw the state transition diagram for the Pattern 'baa' given the input alphabet = {b, a} (6 marks)

ii) Consider the below text

b	b	b	a	b	a	a	a	b	a

Illustrate how you would find the pattern "baa" in the above text using your answer in part f) i) above. (2 marks)

End of Question Paper

Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Information Technology

Final Examination
Year 2, Semester 2 (2024)

IT2070 – Data Structures and Algorithms

Duration: 2 Hours

November 2024

Instructions to Candidates:

- This paper has 4 questions.
- Answer all questions in the booklet given.
- The total marks for the paper is 100.
- This paper contains 7 pages, including the cover page.
- Electronic devices capable of storing and retrieving text, including calculators and mobile phones are not allowed.
- 10 minutes reading time is allowed.

Question 1 **(25 Marks)**

a) What is the key principle by which a stack operates and what are the two main operations associated with it? (1 Mark)

b) How does the stack data structure support real-world applications such as browser history or undo operations in text editors? (2 Marks)

c) Can queues perform the tasks mentioned in (b) more efficiently? Justify your answer. (2 Marks)

d) Consider the following circular queue given in Figure 1. Draw the queue frames and indicate the values for rear, front, and count after performing below operations. (6 Marks)

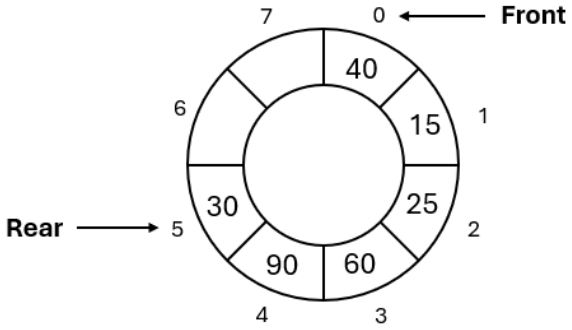


Figure 1. Circular Queue

i) Insert(50)
ii) Insert (70)
iii) Insert (100)
iv) Remove()
v) Remove()
vi) Insert(35)

e) **"A circular queue is considered full when (rear + 1) % size == front"**. Do you agree/disagree with this statement? Justify your answer. (3 Marks)

f) Assume following StackX class given in Figure 2 is implemented. You need to introduce a new method to StackX class to reverse individual words of a given string. Write a Java code segment for ***String reverseString(String input)*** method to take a string input from the user. Use only a stack data structure and obtain the following output. (8 Marks)

Input: "Hello how are you" ***Output:*** "olleH woh era uoy"

StackX
- char[] wordStack - int maxSize - int top
+void push(char c) +char pop( ) +char peek( ) +boolean isEmpty( ) +boolean isFull( )

Figure 2. StackX Class

g) Write the main program to call reverseString(String input) and display the reversed string. ***Hint: You can predefine the size of the stack.*** (3 Marks)

Question 2 **(25 Marks)**

---

a) Differentiate between singly linked lists and doubly linked lists. Give one application for both types. (2 Marks)

b) Consider the linked list given in Figure 3 and execute following code segments sequentially on the linked list. For each code segment display the output.

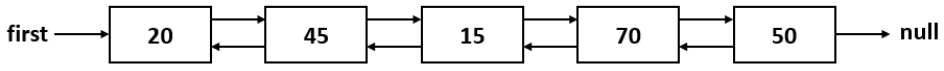


Figure 3. Doubly Linked List

i. (2 Marks)

```
Link temp = first;
while (temp != null)
{
   System.out.print(temp.data + ", ");
   temp = temp.next;
}
```

ii.
```
Link temp = first;
while (temp.next != null)
        temp = temp.next;
while (temp != null)
{
        System.out.print(temp.data + ", ");
        temp = temp.prev;
}
```
(2 Marks)

c) Consider the linked list given in Figure 3. Write down the relevant code segments for the following instances. Assume all operations are sequentially executed on the same linked list.

i. Insert value 30 after the existing node value 70 (4 Marks)

ii. Remove the first link from link list. (2 Mark)

d) Assume a complete binary tree consists of 27 nodes. Calculate the height of the tree. (2 Marks)

e) Consider the values given below. Arrange them into a binary search tree. (2 Marks)

180, 300, 270, 110, 80, 160, 410, 140, 290, 360, 170, 30, 505

f) Consider the tree you have derived in 2)e). Display the elements in the tree in given traversing order.

i. In-order (1 Mark)

ii. Pre-order (2 Marks)

iii. Post-order (2 Marks)

g) Draw the tree structure after performing following operations on the tree you have derived in 2)e).

i. Delete (30) (1 Mark)

ii. Delete (270) (1 Mark)

iii. Delete (180) (2 Marks)

Question 3 **(25 Marks)**

---

a) Compare and contrast Exact Analysis and Asymptotic Notations. (2 Marks)

b) Express T(n) of following pseudocodes using RAM model.

i) (4 Marks)

```
i = 0
for ( j = 0 to 4 )
        while ( i <= 2 )
                i = i + 1
        print j
```

ii) (3 Marks)

```
j = 10
while ( j >= 0 )
        print j
        j = j - 2
```

c) Any recursive problem can have only one base/initial condition. Do you agree with this statement? Briefly explain your answer using an example. (2 Marks)

d) Obtain the recurrence equation for Quick Sort best case. (4 Marks)

```
QUICKSORT (A,p,r)
1 if p < r
2    q = PARTITION(A, p, r)
3    QUICKSORT (A, p, q-1)
4    QUICKSORT (A, q+1, r)
```

e) Solve the recurrence equation received in (d) using Master Theorem. (3 Marks)

$$T(n) = \begin{cases} \Theta\left(n^{\log_b a}\right) & f(n) = O\left(n^{\log_b a - \varepsilon}\right) \rightarrow f(n) < n^{\log_b a} \\ \Theta\left(n^{\log_b a} \lg n\right) & f(n) = \Theta\left(n^{\log_b a}\right) \rightarrow f(n) = n^{\log_b a} \\ \Theta\left(f(n)\right) & f(n) = \Omega\left(n^{\log_b a + \varepsilon}\right) \rightarrow f(n) > n^{\log_b a} \\ & \text{if } af(n/b) \le cf(n) \text{ for c} < 1 \text{ and large } n \end{cases}$$

f) Comment on the performance of (e) comparing with Merge Sort algorithm. (2 Marks)

g) Illustrate the operations of New Insertion sort algorithm for the array A with given set of elements. (For illustration process assign the values only once and then use a diagrammatic approach to reach the answer.) (5 marks)

	1	2	3	4
A	**7**	**1**	**5**	**3**

```
NEW-INSERTION-SORT (A)
1 for j = 2 to A.length
2       i = 1
3       while A[j] > A[i]
4               i = i + 1
5       key = A[j]
6       for k = 0 to j – i – 1
7               A[j-k] = A[j-k-1]
8       A[i] = key
```

Question 4 **(25 Marks)**

---

a) State whether the following statements are TRUE or FALSE. (5 Marks)

i) All heaps are full binary trees.
ii) Heapify() is a recursive algorithm.
iii) Time complexity of build_heap() is O(n).
iv) Time complexities of heapsort() and mergesort() are equal.
v) Priority queue is an application of heaps.

b) Using a suitable example derive the equation for the number of shifts occurred in Naïve String-matching algorithm, where ***n*** is the number of characters in Text and ***m*** is the number of characters in Pattern. (3 Marks)

c) Compute the number of shifts occurred in the following Text and the Pattern using the equation obtained in (b). (2 Marks)

T = abcdefg
P = add

d) Briefly describe the performance of the scenario mentioned in (b) along with its complexity using Big (O) notation. (3 Marks)

e) Following is the pseudocode for heapify() algorithm.

```
MAX_HEAPIFY (A,i)
1.      l  = LEFT(i);
2.      r  = RIGHT(i);
3.      if l ≤ A.heap_size and A[ l ] > A[ i ]
4.              largest = l;
5.      else largest = i;
6.      if r ≤ A.heap_size and A[r] > A[largest]
7.              largest = r;
8.      if largest ≠ i
9.              exchange A[i] with A[largest]
10.             MAX_HEAPIFY (A,largest)
```

i) What is the purpose of having if() condition in line number 3? (2 Marks)

ii) What is the purpose of having if() condition in line number 6? (2 Marks)

f) Draw the state transition diagram for Pattern P = ***bbb*** along with the input alphabet = {a,b}. (8 Marks)

End of the paper

Question 1 **(25 marks)**

a) What is the best data structure that can be used to reverse a given string? State reasons.
(3 marks)

b) Constructor of the stack class is implemented as follows,

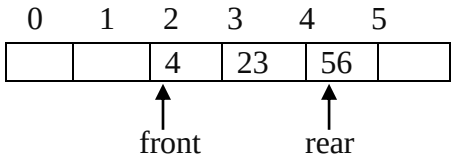
```
public StackX()
{
   stArr = new double[10];
   maxSize = 10;
   top = -1;
}
```

i. State a disadvantage of using the above constructor. (3 marks)

ii. Rewrite the constructor to avoid the disadvantage stated above. (2 marks)

c) Implement the **`size()`** method to return the number of elements in the stack.
(3 marks)

d) Consider the following **circular queue** frame and draw the resultant queue frame after executing following code segment.
(3 marks)




```
q1.insert(66);
q1.insert(56);
q1.insert(q1.remove());
```

e) Consider the given **`remove()`** method implemented for a circular queue. This code contains some errors. Write the line numbers with errors and correct them.

(6 marks)

```
1. public int remove()
2. {
3.      if (rear == -1)
4.      {
5.           System.out.println("Queue is empty");
6.           return false;
7.      }
8.      else
9.      {
10.          int temp = queueArray[front++];
11.          nItems--;
12.          return temp;
13.     }
14.}
```

f) "rear == -1condition can be used to find whether a linear queue is empty" . Do you agree with this statement? Justify your answer.

(5 marks)

Question 2 **(25 marks)**

a) Consider the following LinkedLists and write a code segment that would change the LinkedList from diagram A to diagram B. (First, P1, and P2 are references to given links)

(6 marks)

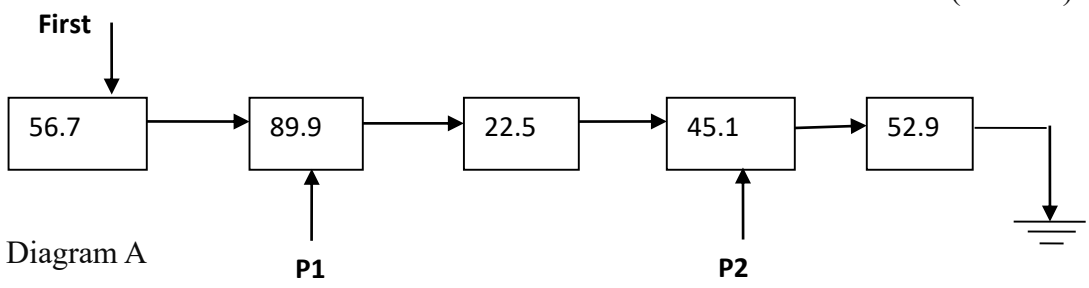


Diagram A

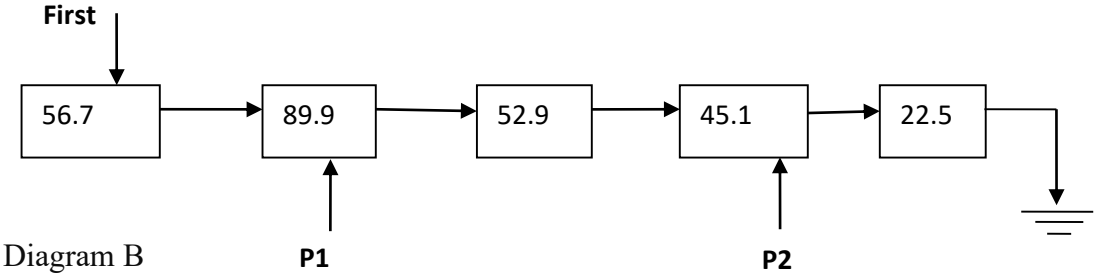


Diagram B

b) A company maintains the items of their store using a linked list. An item consists of

Item Code	integer
Unit Price	double

Assume the **item** class and the **linked list** class has already been implemented.

i) Implement a method called `findUnitPrice()` in the linked list class to return the unit price when the item code is given as a parameter.

(5 marks)

ii) State the modifications to be done to make the above linked list a doubly linked list?

(2 marks)

c) i) Insert the following data in to a Binary Search Tree. (4 marks)

m g k s v i d t

ii) Complete the below method called displayDesc() in the tree class to display the data in descending order. (3 marks)

```
private void displayDesc(Node localRoot)
{
        if (localRoot != null)
        {
                ……
                ……
                ……
        }
}
```

iii) Implement a method to return the maximum value from a binary search tree (5 marks)

Question 3 **(25 marks)**

a) Consider the linear search algorithm given below

```
LINEAR-SEARCH(A, v)
1. for i = 1 to A.length
2.       if A[i] == v
3.           return i
4.   return NIL
```

i) Identify the worst case and best case scenario in linear search algorithm (4 marks)

ii) Find the worst case and best case running time in Big O notation. (2 marks)

b) Consider the merge sort algorithm given below

MERGESORT (A, p, r)

1. **if** p < r
2. q = $\lfloor$(p+r)/2$\rfloor$
3. **MERGESORT** (A, p, q)
4. **MERGESORT** (A, q+1, r)
5. **MERGE** (A, p, q, r)

MERGE(*A*, *p*, *q*, *r*)

1 $n_1 = q - p + 1$
2 $n_2 = r - q$
3 create arrays $L[1..\ n_1 + 1]$ and $R[1..\ n_2 + 1]$
4 **for** $i = 1$ **to** n_1
5 $L[i] = A[p + i - 1]$
6 **for** $j = 1$ **to** n_2
7 $R[j] = A[q + j]$
8 $L[n_1 + 1] = \infty$
9 $R[n_2 + 1] = \infty$
10 $i = 1$
11 $j = 1$
12 **for** $k = p$ **to** r
13 **if** $L[i] \leq R[j]$
14 $A[k] = L[i]$
15 $i = i + 1$
16 **else** $A[k] = R[j]$
17 $j = j + 1$

i) In the MERGE algorithm, what is the purpose of line 8 and 9. Explain the reason for having ∞ (6 marks)

ii) Find the recurrence equation for mergesort algorithm and explain the way you derived it (3 marks)

iii) What is the running time of merge sort? (2 marks)

c) One of the main tasks of the operating system (OS) is to schedule processes for Input Output (IO) devices. When there are several request for IO devices from several processes, OS can create a queue and insert processes into the queue. Assume the queue implemented by the OS is a **priority queue** and numbers are assigned to each process to represent priority with the high number means high priority. Currently there are 8 processes waiting for the IO devices in the priority queue with the priority given below at time t_0.

Process Number	1	2	3	4	5	6	7	8
Priority	150	120	130	24	30	9	4	2

i) Briefly explain which process will get the IO device first? (1 mark)

ii) Which process will get the IO device next? (1 mark)

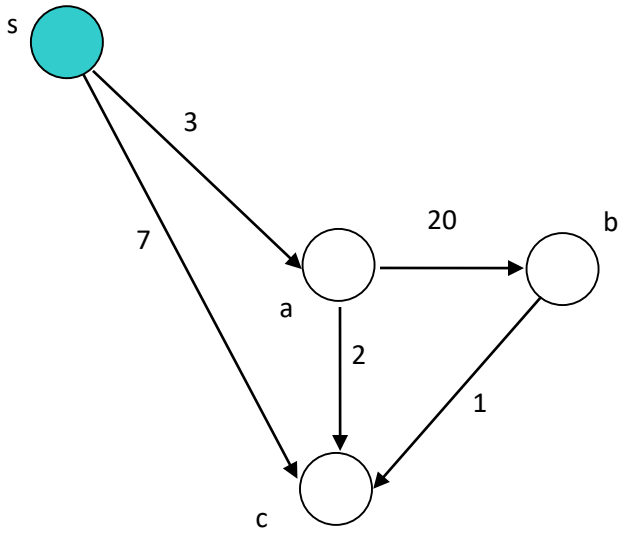
iii) If the new process has come to the queue at the time t_0 with the priority number of 200, Briefly explain how does the OS will insert this process into the queue with the help of Max_**Heap_Insert()** algorithm given.

(6 marks)

```
Procedure  MAX_HEAP_INSERT(A, key)
1. A.heap_size = A.heap_size +1
2. i = A.heap_size
3. while i > 1 and A[PARENT(i)] < key
4.       A[i] = A[PARENT(i)]
5.        i = PARENT(i)
6. A[i] = key
```

Question 4 **(25 marks)**

a) Apply the **Dijkstra's** algorithm given to find the shortest path from the source vertex **s** to all the other vertices of the graph. (For the purpose of illustration, assign the values only once to the given algorithm and use diagrammatic way to reach the answer.)
(6 marks)



INITIALIZE-SINGLE-SOURCE(*G*, *s*)
1 **for** each vertex $v \in G.V$
2 $v.d = \infty$
3 $v.\pi$ = NIL
4 $s.d = 0$

RELAX(*u*, *v*, *w*)
1 **if** $v.d > u.d + w(u, v)$
2 $v.d = u.d + w(u, v)$
3 $v.\pi = u$

DIJKSTRA(*G*, *w*, *s*)
1 INITIALIZE-SINGLE-SOURCE(*G*, *s*)
2 $S = \emptyset$
3 $Q = G.V$
4 while $Q \neq \emptyset$
5 u = EXTRACT-MIN(Q)
6 $S = S \in \{u\}$
7 for each vertex $v \in G.Adj[u]$
8 RELAX(*u*, *v*, *w*)

b)

i) If modulo value is q = 100, how many spurious hits and valid hits do the **Rabin - Karp matcher** encounter in the text T = 10300100200100 when looking for pattern P = 100? (4 marks)

ii) How do you reduce the number of spurious hits in (1)? (2 marks)

iii) What should be the number of spurious hits and valid hits if the best-case scenario occurs in Rabin-Karp algorithm?

(2 marks)

c) Draw the state transition diagram for a string-matching automation for the pattern P = ***abb*** and take the input alphabet as {*a*,*b*}

(6 marks)

d) Following is the **Naïve-String-Matcher** algorithm, which is used to find the occurrence(s) of a pattern string within another string or body of text.

```
Naïve-String-Matcher (T, P)
1.    n = T.length
2.    m = P.length
3.        for s = 0 to n-m
4.            if P[1..m] = T[s+1..s+m]
5.                print "Pattern occurs with shift" s
```

Given the text and pattern as follows;

Text ***T***

a	b	a	b	a	a	a	a	b

Pattern ***P***

a	b	a

i) How many comparisons would occur in this algorithm? (2 marks)

ii) Show that the worst-case time complexity of the above algorithm is $O(m(n-m+1))$ where n is the number of characters in the text and m is the number of characters in the pattern. (3 marks)

End of the Paper