Examination Cover Page

Examination Period:	2019 HE Term 2
Academic Institution:	Central Queensland University
Academic Group:	Higher Education Division
Academic Career:	Postgraduate
Examination Type:	2019 HE Term 2 Standard

Affix Student ID Sticker here

I have read and understood the penalties involved if I do not abide by the rules outlined on the back of this examination paper.

Student Signature: _____ Student ID Number: | | | | | | | | | |

Unit:	Data Structures and Algorithms
Subject Area:	COIT
Catalog Number:	20256
Paper Number:	1
Component:	ALL Components

Duration	180 minutes	Exam Conditions	Open Book
Perusal Time	15 minutes		
First Contact	Mahbub Ahmed	Contact Number	0449599509
Second Contact	Michael Li	Contact Number	(07) 4930 6337

Office Use: Release examination paper via the CQ University Past Exams website two weeks after the DE/SE examination period? Yes

Instructor Authorised/Allowed Materials
Dictionary - non-electronic, concise, direct translation only (dictionary must not contain any notes or comments). No Calculators Permitted

Special Instructions to Students:

Please see instruction sheet on first page of the examination paper.

Examination Office Supplied Materials
1 x Rough Paper 1 x Exam Answer Booklet

Questions Answered		**Marks**	**Questions Answered**		**Marks**

Number of examination answer booklets used:
Number of separate sheets attached (Do not include rough paper):

This examination paper is not to be released to the student at the conclusion of the examination.

Central Queensland University considers improper conduct in examinations to be a serious offence.

Penalties for cheating are exclusion from the University and cancellation with academic penalty from the unit concerned.

Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

Instructions Sheet

1. **Write all answers in the Examination Answer Booklet provided.**

2. **This examination is worth 50 marks. Answer all 10 questions. These are of equal value – 5 marks each.**

3. **Write your answer clearly, use numbered headings or subheadings to show which part of your answer refers to which question. Example: Question 2 (a).**

Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

TOTAL **50 MARKS**

Question 1 **5 Marks**

Given the Linked List operations below, write the output from the executed program.

```
import java.util.LinkedList;
public class Q1StdExam{
   public static void main(String args[]){
   LinkedList<String> aLinkedObj= new LinkedList<String>();
   // Adding elements to the linked list
   aLinkedObj.add("C");
   aLinkedObj.add("B");
   aLinkedObj.addLast("A");
   aLinkedObj.addFirst("D");
   aLinkedObj.add(2, "Q");
   aLinkedObj.add("F");
   System.out.println("Linked list : "+aLinkedObj);
   // Removing elements from the linked list
   aLinkedObj.remove("B");
   aLinkedObj.remove(3);
   aLinkedObj.removeFirst();
   System.out.println("List upon deletion: " + aLinkedObj);
   // Finding elements in the linked list
   boolean status = aLinkedObj.contains("E");
   if(status)
      System.out.println("List contains 'E' ");
   else
      System.out.println("List doesn't contain 'E'");
   aLinkedObj.set(2, "U");
   System.out.println("List upon change : "+aLinkedObj);
  } // end main
} //end Q1StdExam
```

Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

Question 2 **5 Marks**

Write the missing lines of code in the `Q2StdExam` class given below:

a) To add all the values of the `names` array to the `list1` using the Collections method **(2 Marks)**

b) To sort the `names` in `list1` using a Collections method **(1 Mark)**

c) To find the maximum value of `list2` using a Collections method **(1 Mark)**

d) Write the output of the program. **(1 Mark)**

```
import java.util.Arrays;
import java.util.Collections;
import java.util.LinkedList;
public class Q2StdExam {
  public static void main(String[] args) {
    String[]names =
                {"Lily","Li","Sam","Raj","Ron","Sal","Jim"};
    Integer [] ids = {1710,1810,1910,1300};
    LinkedList<String> list1 = new LinkedList<String>();
    LinkedList<Integer>list2 =
                          new LinkedList<>(Arrays.asList(ids));
```

//Write code to add all names to `list1` using a Collections method

//Write code to sort the names in `list1` using a Collections method

//Write code to find the maximum value of `list2`.

```
  System.out.println(list1);
   } //end main
} //end Q2StdExam
```

Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

Question 3 **5 Marks**

Complete the source code given below:

a) To create `treeSet1` of Type `TreeSet` and add the elements of the array named `acroArray` to the `treeSet1` **(2 Marks)**

b) To write code to display the headSet and tailSet based on the String value "FMG" **(2 Marks)**

c) Write output of the program on execution **(1 Mark)**

```
import java.util.*;
public class Q3StdExam{
   public static void main(String[] args){
     String[] acroArray =
                    {"CQU","IBM","TPG","FMG","BIT","CQU"};
```

// Create an object named `treeSet1` of type `TreeSet` and add elements of the //`acroArray` to the `treeSet1`.

//Write code here

//Write code to display the headSet based upon "FMG"

//Write code to display the tailSet based upon "FMG"

```
  } //end main
} // end Q3StdExam
```

Question 4 **5 Marks**

a) By using the definition of Big-O, show that the running time term $T(n)=100+2n+ n^2$ is $O(n^2)$. **(3 Marks)**

b) Consider the code given below which sums the elements of a two dimensional matrix of size N×N looping through each row and for each row all columns, in that matrix. Express the running time to execute this code in Big O notation. **(2 Marks)**

```
int sum =0;
```

// Assume matrix N×N is declared and initialized.

```
for (int i =0; i < N; i++)
     for (int j =0; j < N ; j++)
          sum += matrix[i][j];
```

Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

Question 5 **5 Marks**

Assume that the Country class exists with name, polulation in millions, perCapitaIncome, getter and setter methods and a `toString()` method.
Complete the missing lines of code in the program given below:

a) To display the countries starting with "I" **(2 Marks)**

b) To filter and create a list of income values being greater than 4000 **(2 Marks)**

c) To display the filtered list of values **(1 Mark)**

```
import java.util.*;
import java.util.stream.Collectors;
public class  Q5StdExam{
   public static void main(String[] args) {
   String[]names =
    {"Indonesia","USA","Poland","India","Sweden","Iceland"};
   Integer [] income = {2030, 8030,3050,1230,9030,6030};
```

//Write code to filter and display `names` starting with "I'.

// Write code to filter and create a list of income values being greater than 4000

// Write code to display filtered list of values

```
   } // end main
} // end Q5StdExam
```

Question 6 **5 Marks**

Assume that an sql connection exists to the database named DBRecords; and methods to close the connections exist too. You are given the query named `SQL_CREATE_TABLE_STUDENT` to create a table named Student to store the Student attribute values of a student's Id, name, and email with Id being a primary key and auto incremented.

a) Complete the code to execute the given query **(1.5 Marks)**

b) Complete `PreparedStatement` denoted as `psAddStudent` to insert a student record with given attribute values into the Student table. **3.5 Marks)**

Question 6 (continued over next page)

Question 6 (continued from previous page)

```
import java.sql.*;

public class Q6StdExam {

     private static Statement statement;
     private static String SQL_CREATE_TABLE_STUDENT;
     private static final String MYSQL_URL =
                              "jdbc:mysql://localhost:3306";
     private static final String DB_USERNAME = "mm2root";
     private static final String DB_PASSWORD = "abc123";
     private static final String DATABASE_NAME = "DBRecords";
     private static final String TABLE_NAME = "Student";
     private static boolean tblStudentExists = false;
     private static final String DB_URL =
                                  MYSQL_URL+"/"+DATABASE_NAME;

     public static void main(String[] args){

        Connection dbConn = null;

        PreparedStatement pStmt = null;

     //query string to create Student table
       SQL_CREATE_TABLE_STUDENT =
                        "CREATE TABLE STUDENT IF NOT EXISTS"+
                       "(id INTEGER NOT NULL AUTO_INCREMENT,"+
                       "name VARCHAR(100) NOT NULL," +
                       "email VARCHAR(100) NOT NULL," +
                       "PRIMARY KEY (id))";
     try{

        dbConn=DriverManager.getConnection(DB_URL,
                           DB_USERNAME, DB_PASSWORD);
        statement = dbConn.createStatement();
        //create Student table if not exist
```

// Write code here to execute the given query, `SQL_CREATE_TABLE_STUDENT`

```
        if (!tblStudentExists){
          System.out.println("Table Created!. ");
         }
       }catch(Exception ex) { ex.printStackTrace(); }

} // end main
```

Question 6 (continued over next page)

Question 6 (continued from previous page)

```
public void addStudent(int id, String name,String email){
    PreparedStatement psAddStudent;
```

//Write code here

```
  } //end addStudent method
} //end Q6StdExam
```

Question 7 **5 Marks**

Given the incomplete program of Priority Queue below,

a) Create a `pQueue` object type `PriorityQueue<Integer>`. **(1 Mark)**

b) Add the array elements from `someNumbers` to `pQueue` using appropriate method of PriorityQueue **(2 Marks)**

c) Use appropriate Priority Queue methods to display elements of the `pQueue`, and remove the elements from the `pQueue`. **(2 Marks)**

```
import java.util.PriorityQueue;
public class Q7StdExam{
   public static void main(String[] args){
      int [] someNumbers={7, 4, 9, 1, 3, 2, 5};
```

// Write code here to create a `pQueue` object from `PriorityQueue<Integer>`

//Use appropriate PriorityQueue method to add elements to the `pQueue`
// Write code here

```
      System.out.println("Display and remove from queue: ");
```

//Use appropriate `PriorityQueue` methods to display and remove elements from //the `pQueue`. Write code here

```
   } //end main
}// end Q7StdExam
```

Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

Question 8 **5 Marks**

The following program creates a class named `Q8GenericQueue` and then uses it to create a Queue of integers.

a) Complete the code for the method dequeue which returns a value of type T after testing that the Queue is not empty. **(2.5 Marks)**

b) Write code to remove three values from the Queue with the values from the Array of int, `numbers` given below using your enqueue method. **(2.5 Marks)**

```
public class Q8GenericQueue<T> {
  private LinkedList<T> elements;
  public Q8GenericQueue () {
    elements = new LinkedList<T>();
  }
  public void enqueue (T value){
      elements.offer(value);
   }
```

//Complete code for the `dequeue()` method
//Use appropriate `LinkedList` method to remove elements from the Queue.

```
  public T dequeue () {


  }
  public static void main(String [] args){
    int [] numbers = {23,54,35,63,78,82,91};
    Q8GenericQueue<Integer> queue = new
                                    Q8GenericQueue<Integer>();
    for (int i: numbers)
        queue.enqueue(i);
```

// Write code to display three elements for `queue` using the `dequeue()` method //that you have written.

```
  } // end main
}// end class
```

Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

Question 9 **5 Marks**

Given the incomplete program below to find the recursive summation of any integer number by using the method named `recursiveSum(int aNumber)`. Write the missing lines of code in the source code given below.

a) To write code for the body of the `recusiveSum(int aNumber)` so that it returns the sum of numbers in the range `aNumber` to 1. **(3 Mark)**

b) To write code to call the `recusiveSum(int aNumber)` method to find the sum of numbers 10 to 1 and display the sum. **(2 Marks)**

```
public class Q9StdExam{

  public static int recursiveSum(int aNumber) {
```

//Write code here for the recursive sum that uses recursion to add the numbers //in the rang of `aNumber` to 1**, Not through iteration using for loop**.

```
  } //end recursiveSum

  public static void main(String[] args){
```

//Write code to call the `recusiveSum(int aNumber)` method to find the sum //of numbers 10 to 1 and display the sum

```
  } //end main
} //end Q9StdExam
```

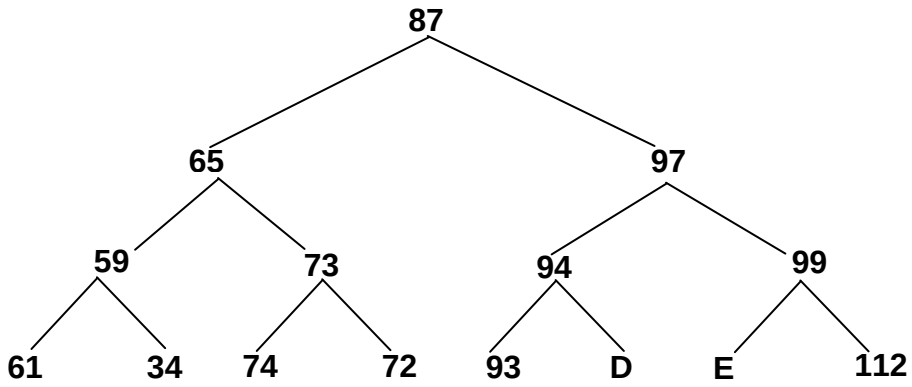
Term 2 Standard examination 2019

Data Structures and Algorithms — COIT20256

Question 10 **5 Marks**

a) In the binary search tree (BST) below, certain values are misplaced and wrong by not following the BST rules. Correct the values. List the values of D and E in the tree below to make it a correct BST. **(2 Marks)**

b) Write the inorder, preorder and postorder traversals of the BST with D and E correctly valued, and properly placed correct values**.** **(3 Marks)**



- End of paper -