MID-SEMESTER EXAMINATION, MARCH-2017
NUMERICAL METHODS (MTH 4002)

Programme: B.Tech
Full Marks: 30

Semester:4th
Time: 2 Hours

Subject/Course Learning Outcome	*Taxonomy Level	Ques. Nos.	Marks
Discuss the different types of computational errors and number systems for machine use	L3	1(a)-(c)	2+2+2
Demonstrate understanding of common numerical methods and how they are used to obtain approximate solutions	L3	2(a)-(c)	2+2+2
Apply numerical methods to obtain approximate real and complex zeros of algebraic and transcendental equations.	L3,L1	3(a)-(c) 4(a)	2+2+2 2
Demonstrate and apply numerical methods to obtain solutions to system of linear equations.	L3L3	4(b),(c)	2+2
Apply difference properties for various functions and use them for construction of polynomials	L3	5(a)-(c)	2+2+2
Construction of polynomials using function values and various order of derivatives	L4		
Application of integration to get approximate solution	L3		

*Bloom's taxonomy levels: Knowledge (L1), Comprehension (L2), Application (L3), Analysis (L4), Evaluation (L5), Creation (L6)

Answer all questions. Each question carries equal mark.

1. (a) Define absolute error and relative error. 2

   (b) Find the corresponding decimal of the binary systems: 2
   $(10101)_2$, $(10.1)_2$.

   (c) Calculate the variance for rounding. 2

2. (a) Find the formula for the root of $ax^2-2bx+c=0$. 2

   (b) Evaluate $\frac{1}{x}-\cot x$ for small value of x. 2

   (c) Use synthetic division to write $p(x)=x^4+6x^3-12x+1$ 2
   as a factor of $x-2$.

3. (a) Find the value of a for which $e^{ax}-x^2=0$ has a double 2
   zero.

   (b) Find a real zero of $\cos x-x=0$ using modified false 2
   position method.

   (c) Write Newton-Raphson scheme to evaluate $\sqrt[3]{7}$. 2

4. (a) Find the real and imaginary part of the complex function 2
   $w=\ln z$.

   (b) Use partial pivot rule to find the solution of the following 2
   system of linear equations.

   $x_1+2x_2-x_3=2,\ 2x_1-x_2+4x_3=5,\ 3x_1-x_2+x_3=3$.

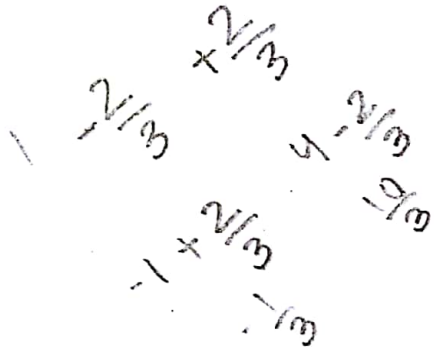
   (c) Use Gauss-Jordan elimination to find the Inverse of the 2
   matrix $\begin{pmatrix}3 & 2\\ 5 & 7\end{pmatrix}$.

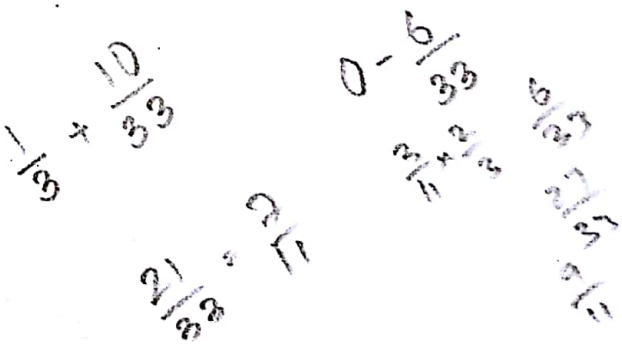
5. (a) Find $\Delta^2(e^x)$ 2

(b) Make a difference table for the sine function with 10° spacing on $0 \le x \le 30^\circ$. 2

(c) Find a polynomial for the data (-1,2), (0,1),(1,2),(2,5) using Newton's formula. 2

End of Questions





$R_2 \to R_2$

$$\left[\begin{array}{ccc|c} 1 & -1/3 & 1/3 & 1 \\ 0 & 1 & -10 & -9 \\ 0 & 7/3 & -2/3 & 1 \end{array}\right]$$

$R_3 \to R_3 - \frac{7}{3}R_2$

$$\left[\begin{array}{ccc|c} 1 & -1/3 & 1/3 & 1 \\ 0 & 1 & -10 & -9 \\ 0 & 0 & -\frac{68}{3} & \end{array}\right]$$

$-\frac{72}{3}$

$-\frac{2}{3} + \frac{70}{3}$

$\frac{72}{3}$

mputer Science and Engineering.
nical Education & Research, S'O'A University

16

Mid-Semester Examination, September-2017
Algorithm Design-II (CSE 4131)

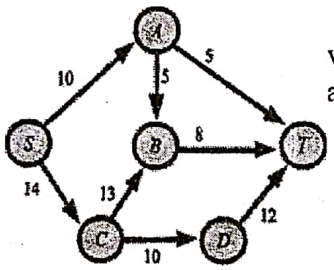
mester: 5th Branch:CSE, CSIT Full mark: 30 Time: 120 Mins.

Subject Learning Outcome	*Taxonomy Level	Ques. No.	Marks
i) understand the network flow problem and apply it to real-world problems.	L3	2(a), 2(b), 2(c)	2+2+2
ii) use a greedy approach to solve an appropriate problem and prove if the greedy rule chosen leads to an optimal solution.	L3, L4, L5,	3(a), 3(b), 3(c)	2+2+2
iii) use recursive backtracking to solve an appropriate problem and identify errors in incorrect implementations.	L3, L4, L5	4(a), 4(b), 4(c)	2+2+2
iv) describe various heuristic problem solving methods.			
v) use dynamic programming to solve an appropriate or provide a recursive solution using memoization.	L3, L4, L5	5(a), 5(b), 5(c)	2+2+2
vi) - distinguish between computationally tractable and intractable problems. - define and relate class-P, class-NP and class NP-complete. - given a problem in NP, define an appropriate certificate and the verification algorithm.			
vii) understand the concept of approximation ratio (with emphasis on constant-factor approximation)			
viii) identify and apply an appropriate algorithmic approach to solve a problem.			
ix) given a numerical problem, explain the challenges to solve it.			

*Bloom's taxonomy levels: Knowledge (L1), Comprehension (L2), Application (L3), Analysis (L4), Evaluation (L5), Creation (L6)

Answer all five questions.

All questions carry equal marks. All bits of each question carry equal marks.

Q1. (a) 

What is the maximum s-t flow in the given graph ? Draw the residual graph you get at the very end of running the Ford-Fulkerson algorithm on this graph. (There is more than one legal answer) 2

(b) Show the minimum cut in the above flow-network problem. 2

(c) Explain briefly why the maximum possible matching is equal to the maximum flow in bipartite matching problem? 2

Q2. (a) Suppose we use the Ford-Fulkerson algorithm to solve a Maximum Bipartite Matching problem. The very first path P found will necessarily have only 3 edges in it, but it is possible for later iterations to find paths of more than 3 edges. Give an example of a bipartite graph where it is possible for one of the paths found to use 9 edges. 2

(b) A matching of an undirected graph G = (V,E) is a set of edges no two of which have a vertex in common. A perfect matching is a matching in which all vertices are matched. Construct a graph G with 2n vertices and n^2 edges such that G has exactly one unique perfect matching. 2

(c) Give the equations/inequalities for capacity constraints and flow conservation constraints in the flow-network problem. 2

3.(a) GREEDY-ACTIVITY-SELECTOR(s, f) 2

1. n ← length[s]
2. A ← $\{a_1\}$
3. i ← 1
4. for m ← 2 to n
5. do if $s_m \geq f_i$
6. then A ← A U $\{a_m\}$
7. i ← m
8. return A

Which line in the above pseudo-code(which we have discussed in the class) checks the mutual-compatibility (i.e. non-overlapping) of two activities? Give the asymptotic running-time of the pseudocode.

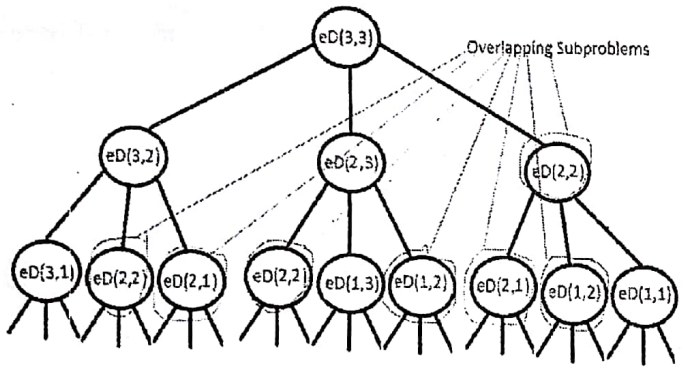
Mid-Semester Examination, September-2017
Algorithm Design-II (CSE 4131)

(b) Find which of the following algorithms are based on Greedy method and give the Greedy-Choice-Property for those algorithms. (i) DFS algorithm, (ii) Kruskal' Minimum Spanning Tree algorithm, (iii) Dijkstra's Shortest-Path algorithm, (iv) Floyd-Warshall's All-Pair-Shortest-Path algorithm.

(c) To design efficient algorithms, sometimes we use a strategy which **always makes the choice that seems to be the best at that moment**. This means that it makes a locally-optimal choice in the hope that this choice will lead to a globally-optimal solution. How this strategy is different from dynamic programming?

Q4 (a) How do we recognize whether a problem admits a dynamic programming based efficient algorithm? How do we further optimize time and space of a dynamic programming based algorithm? 2

(b)



Worst case recursion tree when m = 3, n = 3.
Worst case example str1="abc" str2="xyz"

Given two strings str1='GOD' and str2='MOODY' and edit operations(i.e. insert, delete, replace) that can performed on str1. Find minimum number of edits (operations) required to convert 'str1' into 'str2'.

The worst case happens when none of characters of two strings match. The LHS figure is a recursive call diagram for worst case. Give an asymptotic time complexity for computing the no. of edit operations in the worst case.

(c) Explain how dynamic programming can be used to get a better time complexity in the above question, i.e. Q4(b). 2

Q5. (a) Explain how caching reduces recomputing the same subproblem by constructing the computation tree for computing binomial coefficient C(n,k), i.e. the number of ways choosing k-things out of n-possibilities. 2

(b) Give an efficient pseudocode using dynamic programming to find the longest increasing subsequence within a sequence of n-numbers. 2

(c) Construct the dynamic programming table-M to partition S = {7,3,2,1,5,4,8} into k = 3 or fewer ranges, to minimize the maximum sum over all the ranges, without reordering any of the numbers. 2

****End of Questions****

MID-SEMESTER EXAMINATION, SEPTEMBER-2017

Theory of Computation (CSE 3031)

Semester: 5th **Branch: CSE, CSIT**
Full marks: 30 **Time: 2 hours**

Subject Learning Outcome	*Taxonomy Level	Question Number	Marks
To enhance/develop ability to understand and conduct mathematical proofs for computation and algorithms.	L3	1(a), 3(a)	2+2
To analyse and design finite automata, regular expressions, and regular languages.	L5, L5, L6, L3, L6, L3	1(b), 1(c), 2(a), 2(b), 3(b)	2+2+2 +2+2
To analyse and design pushdown automata, context-free languages, and grammars.			
To analyse and design Turing machines, and recursively enumerable languages.			
To design, implement, and evaluate compuational models to meet desired needs of the languages, and formulate computational models for real-life problems.	L6, L6, L5, L5	2(c), 3(c), 4(a), 4(b), 5(a), 5(b)	2+2+2 +2+2+2
To demonstrate the understanding of key notions, such as algorithm, computability, decidability, and complexity through problem solving.	L6	4(c)	2
To debate, and contrast different classes of formal languages, and their closure properties under language operations.	L6, L5, L3	5(c)	2

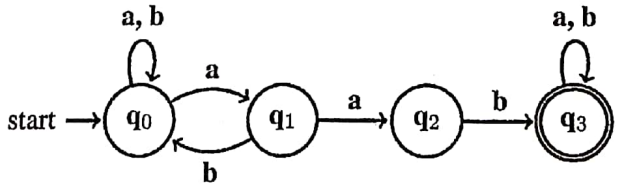
*Blooms taxonomy levels: Knowledge (L1), Comprehension (L2), Application (L3), Analysis (L4), Evaluation (L5), Creation (L6)

Answer all five questions. All questions carry equal marks. All bits of each question carry equal marks.
The figures in the right side indicate marks.

1. (a) A number is rational if it is the ratio of two integers m, and n, ($n \neq 0$). A number is `irrational` if it is not rational. Establish a proof that "$\sqrt{2}$ **is irrational**". **[2]**

   (b) Considering w to be any string of length n over the alphabet {a, b}. Compute the minimum number of states required to construct a finite state machine (FSM) for accepting the language $\{a, b\}^*b\{a, b\}^*\{w\}$. Construct the finite automaton with minimum number of states for $n = 2$. **[2]**

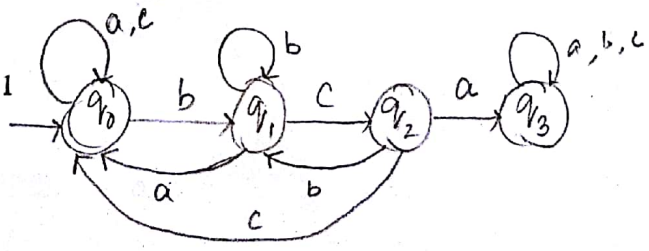
   (c) The transition function for a nondeterministic finite automaton (NFA) can be defined as δ: $Q \times \Sigma \rightarrow P(Q)$. Find the domain and range of δ for the following finite automaton. **[2]**

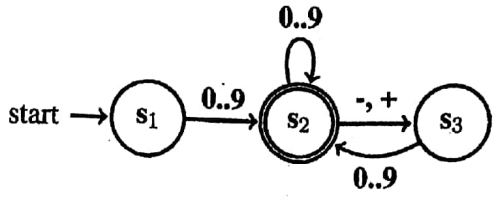


2. (a) Design a deterministic finite automaton (DFA) that recognizes the set of all strings over the alphabet $\Sigma = \{a, b, c\}$ that contains "bca" as substring. **[2]**

   (b) Construct a deterministic finite automaton (DFA) that recognizes the set of all strings over the alphabet $\Sigma = \{a, b\}$ that contains an even number of a's and an odd number of b's. Also give the significance of each state in your design. **[2]**

   (c) For the FSM given below which of these inputs are valid (State with reason):
   (i) 987654321+994-0 (ii) 5-5+2*4 (iii) 9+8+7+6+4+3+2+1 (iv) 0+1+2+1+ (v) 99+88-77





3. (a) Two machines are equivalent if they recognize the same language. Establish a proof that **"every nondeterministic finite automata(NDFA) has an equivalent deterministic finite automata(DFA)".** **[2]**

   (b) Construct a nondeterministic finite automaton (NDFA) that recognizes the set of all strings over the alphabet $\Sigma = \{0, 1\}$ containing a 1 in the second positon from the end (Ex: 00010 and 00011 are in the language but 00100 00101 are not in the language). **[2]**

   (c) Construct an equivalent deterministic finite automaton(DFA) for the nondeterministic finite automaton(NDFA) designed in **Question 3.(b)**. List all the states that are possible during conversion of the NDFA to DFA. Also identify those states which are not reachable from the start state of the constructed DFA. **[2]**

4. (a) An identifier in the **C** programming language is any string of length 1 or more that contains only letters (l), digits (d), and underscore (___) and begins with a letter or an underscore. Let the alphabet given to construct strings is $\Sigma = \{l, d, \_\_\_\}$. For example some valid identifiers over the alphabet are l, ___, l___d, ldd, but dl___, d___l are not. Find the regular expression for the language of all **C** identifiers.**[2]**

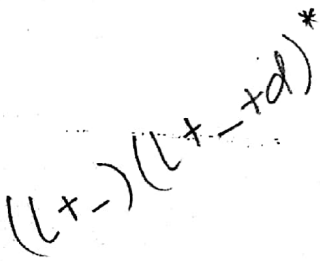
   (b) Construct an NFA-ϵ for the above regular expression using the construction procedure discussed in the Text book. . **[2]**

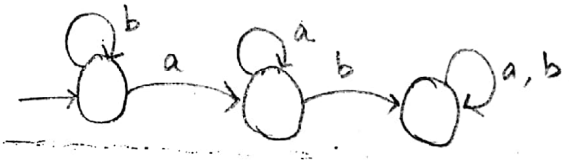
   (c) Convert the above constructed NFA-ϵ to equivalent deterministic finite automaton(DFA) using subset construction. State how many states of the DFA are not reachable from the initial state of the DFA constructed from NFA-ϵ, so that these states are to be deleted. **[2]**

5. (a) The language $L = \{w \mid$ **w contains an even number of a's and odd number of b's and does not contain the substring** $ab\}$ over the alphabet $\Sigma = \{a, b\}$ is the intersection of some simpler languages. In each part, construct deterministic finite automaton(DFA) for the simpler languages, then combine them using the Cartesian product of DFA to get the DFA for the given language **L.** (**Hint:** Use the DFA desined in **Question 2.(b)**) **[2]**

   (b) Give the state diagram of NFA with three states for recognising the language $1^*(001^+)^*$ over the alphabet $\Sigma = \{0, 1\}$. **[2]**

   (c) Find an NFA-ϵ that accepts $\{a\}^*$ over the alphabet $\Sigma = \{a\}$ and is such that if in its state transition diagram a single edge is removed (without any other changes), the resulting automaton accepts $\{a\}^+ = \{a, aa, aaa, ...\}$. Show the removal edge in the design. **[2]**





Department of Computer Science and Engineering
Institute of Technical Education & Research, SOA University

14

Mid-Semester Examination, September-2017
Introduction to Databases (CSE 3151)

Semester: 5th **Branch: CSE, CS&IT**
Full mark: 30 **Time: 2 Hours**

Subject Learning Outcome	*Taxonomy Level	Question No.	Marks
Ability to identify and explain the different components and functionalities of DBMS and their interdependence through the database architecture	L1, L2	1(a),1(b), 1(c), 2(a), 2(c), 3(a), 3(b), 3(c),	16
Ability to use Relational algebra and relational calculus to express queries on relation schemas.	L2, L3		00
Ability to analyze an enterprise schema for given user requirements and apply the conceptual database design principles through ER modeling to construct the ER diagram.	L3, L4	4(a), 4(b), 4(c), 5(a), 5(b)	10
Ability to analyze and design relational database schema using the decomposition and normalization techniques.	L4, L6	5(c)	02
Ability to develop a database using DDL for an organization and apply DML commands to express various queries on the database as per user requirements.	L4, L5	2(b)	02
Ability to recognize and comprehend the functional issues of DBMS like transaction management and database recovery.	L1, L2		

*Bloom's taxonomy levels: Knowledge (L1), Comprehension (L2), Application (L3), Analysis (L4), Evaluation (L5), Creation (L6)

Answer all five questions.

All questions carry equal marks. All bits of each question carry equal marks.

Q1.

a) What is a DBMS? State the advantages of using DBMS over file system? 2

b) Explain the three-schema architecture of DBMS with a neat block diagram and explain the different levels of abstraction. 2

c) What is data independence? Differentiate between logical and physical data independence. 2

Q2.

a) What is a database view? State the advantages of having multiple views for a database. 2

b) Compare DDL with DML. 2

c) Explain the roles and responsibilities of DBA in exerting centralized control over the database. 2

Q3.

a) What is a data model? State the importance of ER model. 2

b) Explain overlapping and disjoint specialization with respect to ER data model. 2

c) Explain participation constraints with respect to ER diagram representation. 2

Q4.

a) Define the following types of keys. 2
Candidate key, Super key, Primary key and Alternate key.

b) Consider an entity set Student (RegNo, Name, Branch, Address, Phone_no, DoB), find out possible *candidate keys, super key, primary key and alternate keys.* 2

c) What is a weak entity? How the primary key of the weak entity set is determined? 2

Q5.

a) Draw an ER-diagram to represent an entity set *Employee (E_Id, Name, Address, Email, Phone_no, Salary, Gross_sal),* where *Address* and *Salary* are composite attributes with sub-attributes *(City, District, State, Pin)* and *(Base_sal, DA, HRA)* respectively. Further, *Email* and *Phone_no* are multivalued attributes and *Gross_sal* is a derived attribute. 2

b) Map the above ER-diagram (Qn. No.-5(a)) to relation schemas. Designate the primary key of each resultant relation schema. 2

c) Illustrate the referential integrity exhibited by the above relation schemas (Qn. No.-5(b)). 2

Department of Computer Science and Engineering.
Institute of Technical Education and Research, SOA University

Mid-Semester Examination, September-2017
Programming Practice-2 (CSE 3042)

Semester: 5^{th} **Branch:** CSE, CSIT
Full mark: 30 **SET-A** **Time:** 120 Mins.

Subject Learning Outcome	*Taxonomy Level	Ques.No.	Marks
use classes to encapsulate data storage structures and the class interface. Searching, insertion, and deletion in arrays and ordered arrays.	L3, L5,L6	Q1	6
understand and demonstrate three simple sorting techniques: the bubble sort, selection sort, and insertion sort.	L3,L5	Q2	6
understand and use three data structures that can be thought of as Abstract Data Types (ADTs): the stack, queue, and priority queue.	L3, L5,L6	Q3	6
understand and use linked list, Designing different problem using linked list.	L3, L5,L6	Q4	6
understand and implement recursion in specific problems such as towers of hanoi and merge sort.	L3, L5,L6	Q5	6
Understand and implement simple tree structure: unbalanced binary search trees and to demonstrate insertion, deletion, and traversal of binary search tree.	L3		
Understand and implement weighted graphs and more complex algorithms involving the minimum spanning trees and shortest paths.	L3		

*Bloom's taxonomy levels: Application (L3), Analysis (L4), Evaluation (L5), Creation (L6)

Answer all five questions. All questions carry equal marks. All bits of each question carry equal marks. Answer to all the bits of a question must be at one place

Q1. A bank wants to automate its customer account, i,e. creating new customer account, and performing some operations like deposit, balance checking, withdraw. Implement the above according the following steps.

Note: Ans of 1.a, 1.b, 1.c should be one program.

(a) Create a class to implement customer account. [2]
(b) Define member functions to perform operations, like deposit, withdraw, balaceChacking. [2]
(c) Create a bank application class to perform operation on bank account. [2]

Q2. Design a class as follows

```
class Array
{
  int arr[];          // reference to an array of integer
  int nEle;           // number of element in the array
  int maxSize;        // maximum size of the array;
   public Array(int max){//define the constructor as in Q2(a)}
   public void insert(int element){// define the function as in Q2(a)}
   public int delete(int element){ // define the function as in Q2(a)}
   public void display(){//  define the function as in Q2(a)}
   public boolean Search(int element)
   {// define the function as in Q(b)}
}
```

Note: Ans of 2.a, 2.b, 2.c should be one program.

(a) Constructor- allocates memory for array of max size, initialize other member variables, insert function- insert an element in the array in sorted sequence, delete function delete an element from array and return its position and return -1 if element not present, display function displays the all array elements. [2]

(b) Search function search an element from array and return true if element is present otherwise return false. [2]

(c) Create another class ArrayApp inside it create an object of Array class and insert some element in the array, display it, delete some elements and search an element from the array of elements and print if it is present or not. [2]

Q3. ArmyX is a government organization concerned with the recruitment of new Officers. They follow a rigorous procedure to hire the new officers. Their rigorous procedure involves finding the best talent and also eliminating the unfit talent during the selection process. ITCX is a software company which responded to ArmyX for providing a solution for their recruitment by providing a software which does adaptive assessment of talent by using an online test. Since the number of candidates is very huge the recruitment spans over a week. Now the task of software is to sort all the candidates in order to mark a cut-off for the selection process. Once the cut off has been finalized they have to mail all the candidates about their results. So they have decided to optimize the process by combining the sorting and mailing process into one. They planned to use Bi-Directional bubble sort so that after the first iteration they will get one selected candidate and one rejected candidate. As soon as a candidate is in its sorted position they will print to screen that a mail is sent. The candidate array holds the mark of ith candidate in ith position of array.
Note: Ans of 3.a, 3.b, 3.c should be one program.

(a) Implement a constructor which allocates memory for holding the candidates mark. Implement an insertion method which handles the insertion into unordered candidate mark array. [2]

(b) Implement the Bi-Directional bubble sort algorithm. Whenever a candidate is in its sorted position print to the screen "Mail sent to Candidate " + indexOfCandidate + marksOfCandidate. [2]

(c) Create object of the above created class and invoke the method as required. [2]

Q4. Create a class InsertionArray having data member an array of integer and add the method to it as follow:
Note: Write one program for Q4. and Q5.

(a) Add a constructor , insert method and display method to it. [2]

(b) Add a method to InsertionArray class which sort array elements using insertion sort. [2]

(c) Add a method to InsertionArray class which remove duplicate from the sorted array in $O(N)$ time, where N is the number of elements in the array [2]

Q5. This question is continued from Q4. :

(a) Another simple sort is the odd-even sort. The idea is to repeatedly make two passes through the array. On the first pass you look at all the pairs of items, $a[j]$ and $a[j+1]$ where j is odd If their key values are out of order, you swap them. On the second pass you do the same for all the even values of j. You do these two passes repeatedly until the array is sorted. Add a method to the InsertionArray class which sort the array using this odd even sort concept. [2]

(b) Modify the insertionSort() method in InsetionArray class so it counts the number of copies and the number of comparisons it makes during a sort and displays the totals. [2]

(c) Create an Application to create the object of the InsertionArray class and invoke the method [2]

.....*End of Question*.....