**Birla Institute of Technology and Science, Pilani, Rajasthan**

**First Semester 2014-15**

**CS F222 / IS F222 / MATH C222 – Discrete Structure for Computer Science**

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

**Start Date: Oct 18, 2014** **Due Date: Oct 28, 2014 by 23:59 Hrs**

**Instructions:**

- **Please read all the instructions carefully before attempting (and submitting the solution by email) the assignment problems.**
- The assignment will be of total **20 marks**.
  - **15 marks** for your program
  - **5 marks** for **correct execution** (will be awarded only if the output comes correct for **all test cases** and the logic of the program is sensible and correct)
- **Every student must attempt a question as per the following strategy. (XXX mod 52) +1 is the question number to be attempted where XXX represents the last three digits of your ID No.**
- All programs must be written in **C language** only. During evaluation, the programs shall be compiled using **gcc compiler** only. Programs compiled on any other variant of C compiler will not be considered.
- All programs must take **input** from a **text (.txt) file** as an **argument** to the program. The format of the input text file must be in the format specified with every question. The test cases from the evaluator's side for evaluation will also be in the same format.
- The **output** must come out as **display over the terminal** or as an **"output.txt"** file as given in the question. The format of the output (if any) has been specified in the question itself. Wherever there is no output format specified, the output should be meaningfully printed.
- Your **deliverable** should be a **zip (.zip) file** with your **ID No as its title**. (E.g. 2013A7PS559P.zip) Your deliverable should contain one or more input files (more details are given in next point) in addition to your program files **(*.c and *.h files)** along with a ***makefile*** that will build/compile your program and generates an executable whose title should be ***"exe"***. For evaluation, this executable shall be used to run your program along with the required input files as arguments (test cases from our side). A sample execution is given below.

  ***make***

  ***./exe input1.txt***
- Your deliverable should also contain one or more input files that you have used as test cases. A **readme.txt** file is also mandatory in the deliverable that would specify the steps to execute your program along with the required arguments if there is any deviation from the steps mentioned above.
- **You must use your BITS email ID only to submit your assignment**. You must email (single email i.e. only one submission) your deliverables to rohil@pilani.bits-pilani.ac.in and cc to jagatsesh@pilani.bits-pilani.ac.in with subject as **DSCS_2014_1_Assignment – IDNo – Name,** where IDNo is your BITS ID.No. and Name is your name as per the course list.
- ***The deadline of submission is October 28, 2014 (Tuesday) by 23:59 hrs. Late submissions shall be entertained up till November 2, 2014 (Sunday) by 23:59 hrs with loss of one mark for every 6 hours (or part of it) of late submission.***

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

- Please note that if there is any data structure/ algorithm specified in the question, coding has to be done as per that data structure /algorithm only. Use of any other data structure / algorithm would straight away fetch zero marks. For example if it has been asked to use Adjacency List to represent Graph, you must use Adjacency List only. You can't use Adjacency Matrix instead. Also inter-conversion of these data structures internally in the program for ease of programming is not permitted. If found so, zero marks will be awarded straight away.
- Programs submitted must be **ORIGINAL**. Original means written by you. If your programs are found to be copied from any online resource, marks awarded will straight away be zero. Also if anyone copies the code from some of his/her felloe student's program, in addition to disciplinary action, marks awarded will be straight away be zero for both the students. Any other sort of plagiarism shall be very seriously dealt with.

**Birla Institute of Technology and Science, Pilani, Rajasthan**

**First Semester 2014-15**

**CS F222 / IS F222 / MATH C222 – Discrete Structure for Computer Science**

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

## Problems Set

1. Given a recurrence relation of the form $a_n + Aa_{n-1} = C.\ F^n$. Given $a_0 = R$. Write a program that takes in the input A, C & R and prints the solution of the recurrence relation. The output should be printed on the terminal in the format –

   *X pow(M,n) + Y pow(L,n) +…….*
   Where X,M,Y, L are any real numbers.

   Sample input file:
   *A=1*
   *C=0*
   *R=2*

2. Given a recurrence relation of the form $a_n + Aa_{n/2} = C$. Given $a_0 = a_1 = C$. Write a program that takes in the input A & C, and prints the solution of the recurrence relation. The output should be printed on the terminal in the format –

   *X pow(M,n) + Y pow(L,n) +…….*
   Where X,M,Y, L are any real numbers.

   Sample input file:
   *A=1*
   *C=0*

3. Given a relation R in the form of tuple – {(x1,y1),(x2,y2),……}. Write a program that takes in the input from a text file in the form mentioned above, parses it and identifies whether this is an equivalence relation or not. The input set could be of any size. The output should be displayed over the terminal.

   Sample input file:
   {(1,2),(3,4),(5,6)}

4. Given a relation R in the form of tuple – {(x1,y1),(x2,y2),……}. Write a program that takes in the input from a text file in the form mentioned above, parses it and computes transitive closure, reflexive closure and symmetric closure for R. The input set could be of any size. The output should be printed on the terminal in the same format.

   Sample input file:
   {(1,2),(3,4),(5,6)}

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

5. Given a relation R in the form of tuple – {(x1,y1),(x2,y2),……}. Write a program that takes in the input from a text file in the form mentioned above, parses it and identifies whether this is a partially ordered set or not. The input set could be of any size. The output should be displayed over the terminal.

   Sample input file:
   {(1,2),(3,4),(5,6)}

6. Given a relation R in the form of tuple – {(x1,y1),(x2,y2),……}. Write a program that takes in the input from a text file in the form mentioned above, parses it and identifies whether this is a lattice or not. The input set could be of any size. The output should be displayed over the terminal.

   Sample input file:
   {(1,2),(3,4),(5,6)}

7. Given a Set S = {x1,x2,x3,x4,x5,….} whose size is n (< 20). Write a program to list all the partial orders (subsets of S) on S with size k. The relation over which the partial order is defined is divisibility. The output should be displayed over the terminal.

   Sample input file:
   {1,2,3,4,5,6,7,8}

   Sample output format:
   {1,2,3,4,6,8}

   The value of k should be taken as input by the program using *scanf()* statement.

8. Given a Set S = {x1,x2,x3,x4,x5,….} whose size is n (< 20). Write a program to list all the partial orders (subsets of S) on S with size k. The relation over which the partial order is defined is “is multiple of”. The output should be displayed over the terminal.

   Sample input file:
   {1,2,3,4,5,6,7,8}

   Sample output format:
   {1,2,4,8}

   The value of k should be taken as input by the program using *scanf()* statement.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

9. Given a Set S = {x1,x2,x3,x4,x5,….} whose size is n (< 20). Write a program to list all the total orders (subsets of S) on S with size k. The relation over which the partial order is defined is divisibility. The output should be displayed over the terminal.

   Sample input file:
   {1,2,3,4,5,6,7,8}

   Sample output format:
   {1,2,4,8}

   The value of k should be taken as input by the program using *scanf()* statement.

10. Given a Set S = {x1,x2,x3,x4,x5,….} whose size is n (< 20). Write a program to list all the total orders (subsets of S) on S with size k. The relation over which the partial order is defined is "is multiple of". The output should be displayed over the terminal.

    Sample input file:
    {1,2,3,4,5,6,7,8}

    Sample output format:
    {1,2,4,8}

    The value of k should be taken as input by the program using *scanf()* statement.

11. Given a Set S = {x1,x2,x3,x4,x5,….} whose size is n (< 20). Write a program to list all the lattices (subsets of S) on S with size k. The relation over which the partial order is defined is divisibility. The output should be displayed over the terminal.

    Sample input file:
    {1,2,3,4,5,6,7,8}

    Sample output format:
    {1,2,4,8}

    The value of k should be taken as input by the program using *scanf()* statement.

12. Given a Set S = {x1,x2,x3,x4,x5,….} whose size is n (< 20). Write a program to list all the lattices (subsets of S) on S with size k. The relation over which the partial order is defined is "is multiple of". The output should be displayed over the terminal.

    Sample input file:
    {1,2,3,4,5,6,7,8}

    Sample output format:
    {1,2,4,8}

    The value of k should be taken as input by the program using *scanf()* statement.

13. Given a partially ordered set S = {x1,x2,x3,x4,x5,….}. Write a program that takes in the input from a text file in the form mentioned above, parses it and partitions the set into partitions containing disjoint anti chains. The partitioning constructed should be of the minimum possible size. The output should be of the form – {(x1,x2,x3); (x5,x8,x4); …..}. where x1, x2, … are any integers.
    The output should be printed on the terminal. Refer to any online material to know more about chains and anti chains in posets.
    Sample input file:
    {1,2,3,4,5,6,7,8}

14. Write a program that takes a graph in adjacency matrix representation with vertices < 16. Determine graph's chromatic number, i.e. minimum number of colors required to label the vertices such that no two adjacent vertices have same color. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    The output should be printed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

15. Given a graph G in the form of Adjacency matrix representation. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Write a Program that calculates the length of the minimal path from vertex *i* to vertex *j* where i & j are any integers. Values of I & j are to be taken as input using *scanf() statement* in the program. The output should be printed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

16. Given a graph G in the form of Adjacency matrix representation. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Write a Program that outputs the Minimal Spanning Tree of G using Kruskal's Algorithm. The output should be printed to terminal in the above format. You can assume that the number of vertices in the whole graph to be less than 10.

17. Write a program that takes two graphs G1 and G2 as inputs in the adjacency matrix format and checks if G1 is a spanning tree of G2. The graphs have to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    The output has to be displayed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

18. Write a program that takes two graphs G1 and G2 as inputs in the adjacency matrix format and checks if G1 is a sub graph of G2. The graphs have to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    The output has to be displayed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

19. Given a graph G in the form of Adjacency matrix representation. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Write a Program that outputs the Minimal Spanning Tree of G using Prim's Algorithm. The output should be printed in the same format as above. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

20. Given a graph G in the form of Adjacency matrix representation. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Write a Program that checks whether the given graph is a Tree or not. The output should be displayed on the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

21. Given a graph G in the form of Adjacency matrix representation. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Write a Program that checks whether the given graph is a height balanced tree or not. You can assume *ith* vertex to be the root of the tree. Value of *i* has to be taken by the program using a *scanf()* statement. The output has to be displayed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

22. Given a graph in adjacency matrix format. Write a program that takes in the graph and checks if the graph has got any cycles in it. The graph have to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    The output has to be displayed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

23. Given a graph G in the form of Adjacency list representation. The graph has to be read from a file in the following format and the adjacency list data structure has to be constructed.

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Write a Program that calculates the length of the minimal path from vertex *i* to vertex *j* where i & j are any integers. Values of i & j are to be taken as input arguments using *scanf() statement* in the program. You can assume that the number of vertices in the whole graph to be less than 10.

24. Given a graph G in the form of Adjacency lsit representation. The graph has to be read from a file in the following format and the adjacency list data structure has to be constructed.

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Write a Program that outputs the Minimal Spanning Tree of G using Kruskal's Algorithm. The output should be printed to terminal in the above format. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

25. Write a program that takes two graphs G1 and G2 as inputs in the adjacency list format and checks if G1 is a spanning tree of G2. The graphs have to be read from a file in the following format and the adjacency list data structure has to be constructed.

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    The output has to be displayed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

26. Given a graph G in the form of Adjacency list representation. The graph has to be read from a file in the following format and the adjacency list data structure has to be constructed. –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Write a Program that outputs the Minimal Spanning Tree of G using Prim's Algorithm. The output should be printed in the above format to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

27. Given a graph G in the form of Adjacency list representation. The graph has to be read from a file in the following format and the adjacency list data structure has to be constructed. –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Write a Program that checks whether the given graph is a Tree or not. The output has to be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

28. Given a graph G in the form of Adjacency list representation. The graph has to be read from a file in the following format and the adjacency list data structure has to be constructed. –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Write a Program that checks whether the given graph is a height balanced tree or not. You can assume *ith* vertex to be the root of the tree. Value of *i* has to be taken by the program using a *scanf()* statement. The output has to be displayed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

29. Write a program that takes two graphs G1 and G2 as inputs in the adjacency list format and checks if G1 is a sub graph of G2. The graphs have to be read from a file in the following format –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    The output has to be displayed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

30. Given a graph in adjacency list format. Write a program that takes in the graph and checks if the graph has got any cycles in it. The graph have to be read from a file in the following format –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    The output has to be displayed over the terminal. 31.You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

31. Write a program that takes a graph in adjacency list representation with vertices < 16. Determine graph's chromatic number, i.e. minimum number of colors required to label the vertices such that no two adjacent vertices have same color. The graph has to be read from a file in the following format –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    The output should be printed over the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

32. Given a graph in Adjacency Matrix format. Write a program which counts the number of disjoint paths between any two given ($i^{th}$ *and* $j^{th}$) vertices. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Values of i & j are to be taken as input using *scanf() statement* in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

33. Given a graph in Adjacency List format. Write a program which counts the number of disjoint paths between any two given ($i^{th}$ *and* $j^{th}$) vertices. The graph has to be read from a file in the following format –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Values of i & j are to be taken as input using *scanf() statement* in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

34. Given a weighted graph in Adjacency Matrix format. Write a program to find all the pairs of vertices, for which the sum of the weights of the edges of any path between them is K. The graph has to be read from a file in the following format –

    Sample input file:
    0,2,3,0,0
    2,0,0,4,5
    3,0,0,0,0
    0,4,0,0,0
    0,5,0,0,0

    Value of K has to be taken as input argument using *scanf()* statement in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

35. Given a weighted graph in Adjacency Matrix format. Write a program to find all edge-disjoint equi-cost paths in the graph between a pair of vertices ($i^{th}$ & $j^{th}$). The graph has to be read from a file in the following format –

    Sample input file:
    0,2,3,0,0
    2,0,0,4,5
    3,0,0,0,0
    0,4,0,0,0
    0,5,0,0,0

    Values of i & j are to be taken as input using *scanf() statement* in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

36. Given a graph in Adjacency Matrix format. Given a starting i and ending node j, write a program to find the Shortest Path which covers all the vertices of the graph. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Values of i & j are to be taken as input using *scanf() statement* in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

37. Given a graph in Adjacency List format. Given a starting i and ending node j, write a program to find the Shortest Path which covers all the vertices of the graph. The graph has to be read from a file in the following format –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Values of i & j are to be taken as input using *scanf() statement* in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

38. Given a graph in adjacency matrix format where every edge is labeled with a color – either orange or black. If the edge is colored orange it is indicated by weight 2 in the adjacency matrix. If the edge is colored black it is indicated by weight 3 in the adjacency matrix. Write a program to find a spanning tree for the entire graph that contains exactly k orange edges (or report that no such spanning tree exists). The graph has to be read from a file in the following format –

    Sample input file:
    0,2,3,0,0
    2,0,0,2,3
    3,0,0,0,0
    0,2,0,0,0
    0,3,0,0,0

    Value of k is to be taken as input argument using the *scanf()* statement in the program. The output of the spanning tree (if exists) should be printed to the terminal in adjacency matrix format similar to that of the input file. Else, report that no such spanning tree exists. You can assume that the number of vertices in the entire graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

39. Given a graph in Adjacency Matrix format. Write a program to compute whether two given vertices of the graph (vertex i and vertex j) are adjacent or not. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    Values of i & j are to be taken as input using *scanf() statement* in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

40. Given a graph in Adjacency List format. Write a program to compute whether two given vertices of the graph (vertex i and vertex j) are adjacent or not. The graph has to be read from a file in the following format –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    Values of i & j are to be taken as input using *scanf() statement* in the program. The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

41. Given a graph in adjacency matrix format. Write a program to determine whether it is connected and find the number of connected component if it is not connected. The graph has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

42. Given a graph in adjacency list format. Write a program to determine whether it is connected and find the number of connected component if it is not connected. The graph has to be read from a file in the following format –

    Sample input file:
    1: 4,5,3
    2: 3,4
    3: 2,1,5
    4: 1,2,5
    5: 1,3,4

    The output should be printed to the terminal. You can assume that the number of vertices in the whole graph to be less than 10.

43. Given a positive integer n (< 20), write a program to list all the permutations of the set {1,2,3,….,n} in lexicographic order. The input *n* has to be read by the program using *scanf()* statement. Output has to be printed to the terminal.

44. Given a matrix representing a relation on a finite set S of size less than 10. Write a program to find the matrix representing the transitive closure of this relation using Warshall's algorithm. The matrix has to be read from a file in the following format –

    Sample input file:
    0,1,1,0,0
    1,0,0,1,1
    1,0,0,0,0
    0,1,0,0,0
    0,1,0,0,0

    The output of the program must be displayed on the terminal in a format similar to that of the input file.

45. Write a program to find number of different spanning trees of $k_n$ for given input n (<10). Use adjacency matrix as the underlying data structure to store $k_n$. The input n has to be taken using *scanf()* statement. The output should be displayed on the terminal.

46. Write a program to find number of different spanning trees of $k_n$ for given input n (<10). Use adjacency list as the underlying data structure to store $k_n$. The input n has to be taken using *scanf()* statement. The output should be displayed on the terminal.

**Assignment Problem (Open Book/Notes and Take-Home)** **Maximum Marks: 20 (10% Weight)**

47. A small post office has only 4-cent stamps, 4-cent stamps, 6-cent stamps, 10-cent stamps. Write a program to find the total number of ways to form postage of n (<1000) cents with these stamps. The input *n* has to be taken using *scanf()* statement. The output should be displayed on the terminal.

48. Suppose you are given an unlimited supply of red, blue, and green cards. Write a program to find the total number of ways to form a stack consisting of *n* colored cards in which a green card is never directly on top of another green card. The input *n* has to be taken using *scanf()* statement. The output should be displayed on the terminal.

49. If an unlimited supply of indistinguishable pennies, indistinguishable nickels, indistinguishable dimes, and indistinguishable quarters is available, write a program to find the total number of distinct arrangements of coins that can be formed whose sum is n cents. The input *n* has to be taken using *scanf()* statement. The output should be displayed on the terminal.

50. Write a program to divide square root of the numbers from 1 to n into two sets A and B such that the sum of numbers in Set A is as close as possible to the sum of the numbers in set B. The input *n* has to be taken using *scanf()* statement. The output should be displayed on the terminal.

51. Given positive integer n, write a program to list and count all the bit sequences of length n that don't have a pair of consecutive 0s. The input *n* has to be taken using *scanf()* statement. The output should be displayed on the terminal.

52. Write a program to find whether a point P (x,y) is in the interior or exterior of simple polygon. The polygon is represented by a set of ordered vertices whose order is same as that of edges. For example – polygon {(x1,y1),(x2,y2),(x3,y3)} is a triangle where first vertex is connected to the second, second to the third and third to the first. You will need to take the coordinates of the vertex of the polygon in the above mentioned format from an input file. The coordinates of point P should be taken using *scanf()* statement. The output has to be displayed over the terminal.

    Sample input file:
    {(x1,y1),(x2,y2),(x3,y3)}