Discrete Structures - Assignment

Paper Code – 32341202

BSc (Hons) Computer Science CBCS

Semester – II

Anshul Verma

Roll No – 19/78065

19/78065

Assigment cum practical evaluation

Q.1(a) Among 100 students, 32 study mathematics, 20 study physics, 45 study biology, 15 study mathematics and bialogy, 7 study mathematics and physics, 10 study physics and biology and 30 do not study any of the subjects.

(i) Find the number of students studying all three subjects.

(ii) Find the number of students studying exactly one of three subjects.

Sol: Given, total no. of students, $|S| = 100$

no. of students studying maths, $|M| = 32$

no. of students studying physics, $|P| = 20$

no. of students studying biology, $|B| = 45$

$|M \cap B| = 15$, $|M \cap P| = 7$, $|P \cap B| = 10$

$\therefore |M \cup P \cup B| = 100 - 30 = 70$

(i) $\because$ $|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$ for A, B, C

$\therefore$ Here, $|M \cup P \cup B| = 32 + 20 + 45 - 15 - 7 - 10 + |M \cap P \cap B|$

$70 = 97 - 32 + |M \cap P \cap B|$

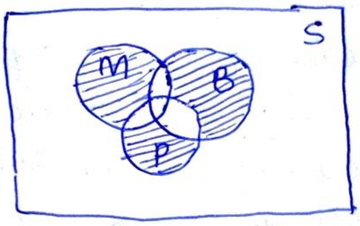
$\therefore |M \cap P \cap B| = 70 - 65 = 5$

$\therefore$ no. of students studying all three subjects $= 5$.

(ii) number of students, studing exactly one of 3 subjects,

$$= (|M| - |M\cap B| - |M\cap P| + |M\cap P\cap S|)$$
$$+ (|P| - |M\cap P| - |P\cap B| + |M\cap P\cap S|)$$
$$+ (|B| - |M\cap B| - |P\cap B| + |M\cap P\cap S|)$$

{from Venn Diagram}



$$= |M|+|P|+|B| - 2(|M\cap B|+|M\cap P|+|P\cap B|) + 3(|M\cap P\cap S|)$$

$$= 32+20+45 - 2(15+7+10) + 3(5)$$

$$= 97 + 15 - (32)\times 2 = 112 - 64$$

$$= \underline{\underline{48}}$$

Q.1 (b) Let R be a binary relation on the set of all positive integers such that,

$$R = \{(a,b) \mid a-b \text{ is an odd +ve integer}\}$$

Explain which of the following is true on R and why?

(i) Reflexive
(ii) Symmetric
(iii) Antisymmetric
(iv) Transitive

Sol: We know that, for positive integers,

even − odd = odd,
odd − even = odd,
even − even = even,
odd − odd = even.

(i) Hence, $(a,a) \notin R$ as $(a-a=0)$ is not a odd positive integer,

$\therefore$ R is not reflexive.

(ii) If $(a,b) \in R$ then $(a-b)$ = odd positive integer, that is $(b-a)$ = odd negative integer.

$\therefore$ If $(a,b) \in R$ then $(b,a) \notin R$, hence R is not symmetric.

(iii) from (ii), if $(a,b) \in R$ then $(b,a) \notin R$,

$\therefore$ R is antisymmetric.

(iv) If $(a,b) \in R$ and $(b,c) \in R$ then $(a-b) =$ odd positive integer and $(b-c) =$ odd positive integer.

$\therefore$ One of a and b is odd +ve and other is negative.

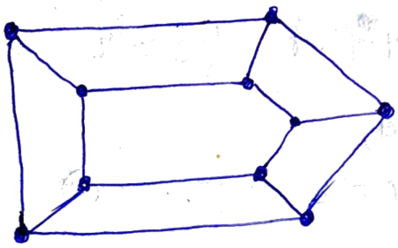
Similarly, one of b and c is odd +ve and other is negative.

If b is odd then a and c have to be even, and then, $(a,c) \notin R$.

If b is even, then a and c have to be odd, and then $(a,c) \notin R$.

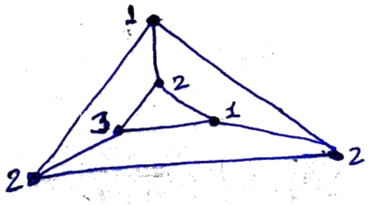
Hence, R is not transitive.

Q.1(c) (i) Derive an expression for the chromatic number of $C_{n,n}$ where $n \geq 3$. $C_{n,n}$ is a graph with two concentric cycles and n vertices connected as shown below:



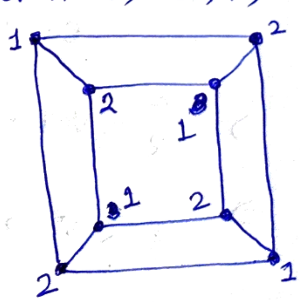
SA:

$\Rightarrow$ for $n=3$, $C_{3,3}$:



$\therefore$ $\chi(C_{3,3}) = 3$

$\Rightarrow$ for $n=4$, $C_{4,4}$:



$\therefore$ $\chi(C_{4,4}) = 2$

$\Rightarrow$ ∴ Generally, * If n is odd & $n \geq 3$, $\chi(C_n, n) = 3$ as it is necessary to alternate two colors as the cycles are individually traversed clock wise (first to $(n-1)^{th}$). When n^{th} vertex is reached, we must use third color.

$\Rightarrow$ If n is even & $n > 3$, $\chi(C_n, n) = 2$, as we can alternate two colours as the cycles are individually traversed clockwise (first to n^{th}). Traversal and coloring of the outer cycle should start with the new vertex, not adjacent to the starting point of traversal of inner cycle.

Hence,

$$\boxed{\chi(C_n, n) = \begin{cases} 2, & n \text{ is even} \\ 3, & n \text{ is odd} \end{cases} \quad \text{for } n \geq 3}$$

Q. 1 (c) (ii) Prove that if a connected graph planar simple graph has e edges and v vertices with $v \geq 3$ and no circuits of less than 3, then $e \leq 2v - 4$.

Sol: ∵ sum of degrees of regions is equal to twice the number of edges,

But each region must have degree ≥ 4 because we have no circuits of length 3.

[∵ given $v \geq 3$]

So, we have,

$$2e \geq 4r \quad \text{or} \quad \frac{1}{2}e \geq r \qquad \text{——— (1)}$$

from Euler's formula,

$$v - e + r = 2 \qquad \text{——— (2)}$$

from (1) & (2),

$$e - v + 2 \leq \frac{1}{e}$$

$$\Rightarrow \frac{e}{2} - v + 2 \leq 0$$

$$\Rightarrow \frac{e}{2} \leq v - 2 \quad \Rightarrow \quad \boxed{e \leq 2v - 4}$$

Hence Proved.

Q.1 (d) (i) Find numberic function for the given generating function.

$$A(Z) = \frac{Z^4}{1-2Z}$$

Sol: Suppose, $a = 2^r$ or $A(z) = \frac{1}{1-2z}$

then, $z^4\left(\frac{1}{1-2z}\right) = S^4 a_r$

$\Rightarrow z^4(2^0 z^0 + 2^1 z^1 + 2^2 z^2 + \ldots\ldots)$

$= 2^0 z^4 + 2^1 z^5 + 2^2 z^6 + \ldots\ldots$

$\therefore$ $a_r = (2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, \ldots\ldots)$

$r = \;\; 0 \;\; 1 \;\; 2 \;\; 3 \;\; 4 \;\; 5 \;\; 6$

$S^4 a_r = (0, 0, 0, 0, 2^0, 2^1, 2^2, \ldots\ldots)$

$r = \;\; 0 \;\; 1 \;\; 2 \;\; 3 \;\; 4 \;\; 5 \;\; 6$

Thus, $S^4 a_r = \begin{cases} 0, & 0 \leq r \leq 3 \\ 2^{r-4}, & r \geq 4 \end{cases}$

Q.1 (d) (ii) Let a_r, b_r, c_r be three numeric functions such that $a_r * b_r = c_r$, Determine b_r.

$a_r = \begin{cases} 1 & r=0 \\ 2 & r=1 \\ 0 & r \geq 2 \end{cases}$ $\quad c_r = \begin{cases} 1 & r=0 \\ 0 & r \geq 1 \end{cases}$

Sol:

$c_0 = a_0 b_0$

$1 = 1 \cdot b_0 \Rightarrow b_0 = 1$

$c_1 = a_0 b_1 + a_1 b_0$

$0 = 1 \cdot b_1 + 2 \cdot 1 \Rightarrow b_1 = -2$

$c_2 = a_0 b_2 + a_1 b_1 + a_2 b_0$

$0 = b_2 + 2(-2) + 0 \cdot 1$

$0 = b_2 - 4$

$\Rightarrow b_2 = 4$

$C_3 = a_0 b_3 + a_1 b_2 + a_2 b_1 + a_3 b_0$

$0 = 1 \cdot b_3 + 2 \cdot 4 + 0 \cdot (-2) + 0 \cdot 1$

$0 = b_3 + 8 + 0 + 0 \quad \Rightarrow \quad b_3 = -8$

$\therefore \quad b_r = \{(-2)^r \quad \because [b = (1, -2, 4, -8, \ldots)]$

Q.1 (e) Solve the following using Master's theorem if applicable. Give reasons if not applicable.

(i) $T(n) = 125 T\left(\frac{n}{5}\right) + n^3$

Sol: Comparing it with, $T(n) = aT\left(\frac{n}{b}\right) + f(n)$,

we get, $a = 125$, $b = 5$, $f(n) = n^3$

$\therefore$ Now, $n^{\log_b a} = n^{\log_5 125} = n^3 = f(n)$

[case 2]

$\therefore T(n) = \Theta(n^{\log_b a} \cdot \log n)$ where $\varepsilon = 0$.

$\therefore T(n) = \Theta(n^3 . \log n)$

(ii) $T(n) = 2T\left(\frac{n}{2}\right) + n \log_2 n$

Sol: Comparing it with $T(n) = aT\left(\frac{n}{b}\right) + f(n)$

we get, $a = 2$, $b = 2$, $f(n) = n \log_2 n$

$\therefore$ Now, $n^{\log_b a} = n^{\log_2 2} = n \neq n \log_2 n \; [f(n)]$

$\therefore f(n)$ i.e. $n \log_2 n$ is asymptotically larger than n but it is not polynomically larger than n. Hence, Master theorem can not be applied to this problem.

Q.1(f) Use insertion sort to sort 7, 4, 6, 3, 2, 5, 9, 8 in ascending order showing the list obtained after each step.

Index	Elements	I itr.	II itr.	III itr	IV itr.	V itr.	VI itr	VII itr.
0	7	4	4	3	2	2	2	2
1	$\boxed{4}$	7	6	4	3	3	3	3
2	6	$\boxed{6}$	7	6	4	4	4	4
3	3	3	$\boxed{3}$	7	6	5	5	5
4	2	2	2	$\boxed{2}$	7	6	6	6
5	5	5	5	5	$\boxed{5}$	7	7	7
6	9	9	9	9	9	$\boxed{9}$	9	8
7	8	8	8	8	8	8	$\boxed{8}$	9
no. of Comparisons →	$\boxed{1}$	$\boxed{1}$	$\boxed{3}$	$\boxed{4}$	$\boxed{1}$	$\boxed{0}$	$\boxed{1}$	

Q.1(g) "If there was a ball game, then ~~lwat~~ travelling was difficult. If they arrived on time, then travelling was not difficult. They arrived on time. Therefore, there was no ball game." Show that these statements constitute a valid arguement.

Sol: Let, B: There was a ball game.

T: Travelling was difficult.

A: Arrived on time.

Premises → $B \to T$, $A \to \neg T$, A

Conclusion → $\neg B$

∴

Step	Statement	Rule
1.	A	P
2.	$A \to \neg T$	P
3.	$\neg T$	T, 1, 2, I_{11} $\{(P, P \to q) \Rightarrow q\}$
4.	$B \to T$	P
5.	$\neg B$	T, 3, 4, I_{12} $\{(\neg q, P \to q) \Rightarrow \neg P\}$

Hence, the arguement is valid.

Q.2(a) Determine whether the function f is a bijection from R to R. Find fog and gof for f and g where,

$$f(x) = 2x^2+3 \text{ and } g(x) = x+1$$

Is $fog = gof$?

Sol:

$\Rightarrow$ Let $x, y \in R$

Assume, $f(x) = f(y)$

$$2x^2+3 = 2y^2+3$$

$$x^2 = y^2$$

$$\therefore x = y$$

$\Rightarrow$ f is injective ——— (1)

$\Rightarrow$ Let $y \in R$. we want to show $y = f(x)$

$$y = 2x^2+3$$

$$y-3 = 2x^2$$

$x = \sqrt{\frac{y-3}{3}}$ which is true iff $y \in [3,\infty)$

$\therefore \Rightarrow f$ is not surjective. ——— (2)

From (1) & (2), $\Rightarrow$ f is not a bijection.

Now, $fog = f(g(x)) = f(x+1) = 2(x+1)^2+3$

$= 2(x^2+2x+1)+3$

$= 2x^2+4x+2+3 = 2x^2+4x+5$

$gof = g(f(x)) = g(2x^2+3) = 2x^2+3+1$

$= 2x^2+4$

$= 2(x^2+2)$

$\therefore \quad fog \neq gof$

Q.2 (b) Eight chairs are numbered from 1 to 8. Two women and three men are to occupy ~~each~~ one chair each. first, the women choose the chairs from amogst the chairs 1 to 4 and then men select from remaining chairs. Find the number of possible arrangements.

Sol: Here, Let the women are W_1 and W_2.
& men are M_1, M_2 and M_3.

$\therefore$ W_1 can take chairs marked from 1 to 4 in 4 diff. ways.
W_2 can take 3 chairs maked from 1 to 4 in 3 diff ways.

$\therefore$ So, total number of ways in which women can take seat, $^4P_2 = \frac{4!}{(4-2)!} = \frac{4\times3\times\cancel{2!}}{\cancel{2!}} = 12$

Now, there will be 6 chairs remaining.

M_1 can take any of 6 chairs in 6 diff. ways.
M_2 can take any of remaining 5 chairs in 5 diff. ways
M_3 can take any of remaining 4 chairs in 4 diff. ways.

$\therefore$ total no. of ways in which men can take seat,

$$^6P_3 = \frac{6!}{(6-3)!} = \frac{6\times4\times5\times\cancel{3!}}{\cancel{3!}} = 120$$

Hence, total number of ways in which men and women can be seated, $^4P_2 \times {}^6P_3 = 12\times120$
$= 1440$.

Q.3 Evaluate the sum: $\sum_{k=1}^{\infty}(2k+1)x^{2k}$

$$\Rightarrow \underbrace{\sum_{k=1}^{\infty}(2k\cdot x^{2k})}_{A} + \underbrace{\sum_{k=1}^{\infty}(x^{2k})}_{B}$$

$$\therefore A \Rightarrow \sum_{k=1}^{\infty} 2k \cdot x^{2k}$$

$$= \sum_{k=0}^{\infty} 2k \cdot x^{2k} - 2 \cdot 0 \cdot x^{2(0)}$$

$$= 2\sum_{k=0}^{\infty} k\,(x^2)^k$$

$$A \Rightarrow \frac{2x^2}{(1-x^2)^2} \qquad \left[\sum_{k=0}^{\infty} k \cdot x^k = \frac{x}{(1-x)^2}\right]$$

$$B \Rightarrow \sum_{k=1}^{\infty} x^{2k}$$

$$= \sum_{k=0}^{\infty} x^{2k} - x^{2(0)} = \sum_{k=0}^{\infty} (x^2)^k - 1$$

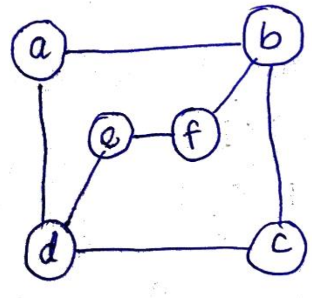
$$= \frac{1}{1-x^2} - 1 = \frac{\cancel{1}-\cancel{1}+x^2}{1-x^2} = \frac{x^2}{1-x^2}$$

$$\left[\sum_{k=0}^{\infty} x^k = \frac{1}{1-x}\right]$$

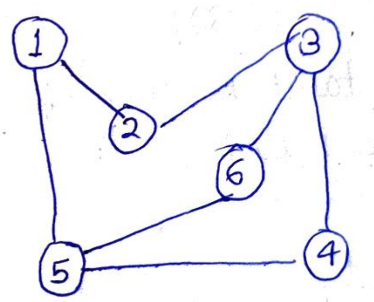
$$\therefore \sum_{k=1}^{\infty} (2k+1)\,x^{2k} = A + B \Rightarrow \frac{2x^2}{(1-x^2)^2} + \frac{x^2}{1-x^2}$$

$$\Rightarrow \frac{3x^2 - x^4}{(1-x^2)^2} \Rightarrow \frac{x^2(3-x^2)}{(1-x^2)^2}$$

Q.4 Determine whether the graphs G and H are isomorphic or not?



G



H

We see that,

$\Rightarrow$ Both the graphs G and H have same no. of vertices and edges.

$\Rightarrow$ Graph G and H both are having 6 vertices and 7 edges each.

$\Rightarrow$ Now, for considering degree of vertices of graph G and H.

$$
\begin{array}{ccc}
\underline{G} & & \underline{H} \\
\deg(a) & = & \deg(6) = 2 \\
\deg(b) & = & \deg(3) = 3 \\
\deg(c) & = & \deg(4) = 2 \\
\deg(d) & = & \deg(5) = 3 \\
\deg(e) & = & \deg(1) = 2 \\
\deg(f) & = & \deg(2) = 2
\end{array}
$$

Now, mapping the vertices,

$$
\begin{array}{c}
a \longleftrightarrow 2 \\
b \longleftrightarrow 3 \\
c \longleftrightarrow 4 \\
d \longleftrightarrow 5 \\
e \longleftrightarrow 1 \\
f \longleftrightarrow 2
\end{array}
$$

Now checking adjacency matrices,

★ Graph G

$$
\begin{array}{c}
\\ a \\ b \\ c \\ d \\ e \\ f
\end{array}
\begin{array}{c}
\begin{array}{cccccc} a & b & c & d & e & f \end{array} \\
\begin{bmatrix}
0 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

★ graph H

$$
\begin{array}{c}
\\ 6 \\ 3 \\ 4 \\ 5 \\ 1 \\ 2
\end{array}
\begin{array}{c}
\begin{array}{cccccc} 6 & 3 & 4 & 5 & 1 & 2 \end{array} \\
\begin{bmatrix}
0 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

Here, both the adjacency matrices for G and H are same.

Thus, <u>Graphs G and H are isomorphic</u>.

Q.5 (a) (i) Prove that if an m-ary tree of height h has l leaves then,

$$h \geq \lceil \log_m l \rceil$$

Sol:

We know that, there are at most m^h leaves in an m-nary tree of height h.

$\therefore \quad l \leq m^h$

taking $\log_m$ on both sides,

$$\log_m l \leq \log_m (m^h) \Rightarrow \log_m l \leq h$$

Because, h is an integer we have

$$h \geq \lceil \log_m l \rceil$$ Hence proved

(ii) If a tree has 2n vertices of degree 1, 3n vertices of degree 2 and n vertices of deg-3, find no. of vertices and edges in the tree.

Sol.

from above, total no. of vertices $= 2n + 3n + n$
$= 6n$

$\therefore$ total no. of edges = total no. of vertices $- 1$
$= 6n - 1$

$\therefore \; 2(6n - 1) = 1(2n) + 2(3n) + 3(n)$

$12n - 2 = 2n + 6n + 3n$

Now, using handshaking theorem,

$$\sum_{v \in V} \deg(v) = 2|E|$$

$\therefore$ ~~$2 6n + 6n + 3n = 2(6n-1)$~~

$$(2n \cdot 1) + (3n \cdot 2) + (n \cdot 3) = 2|E|$$

$$2n + 6n + 3n = 2(6n-1)$$

$$11n = 12n - 2$$

$$\therefore \boxed{n = 2}$$

$\therefore$ total no. of vertices in the tree $= 2\times2 + 3\times2 + 2 = \underline{\underline{12}}$

and total no. of edges in the tree

$= 6\times2 - 1$

$= \underline{\underline{11}}$

<u>Q.5 (b)</u> Is $2^{n+1} = O(2^n)$? Is $2^{2n} = O(2^n)$?

Justify the answer.

Sol: <u>for $2^{n+1} = O(2^n)$</u>; Here, ~~$2^{n+1} \leq c\,2^n$~~

~~$2^{n+1} = 2 \cdot 2^{n+1} \leq c \cdot 2^n$~~

$\therefore$ ~~Let~~ $2^{n+1} = O(2^n)$

$$2^{n+1} \leq c \cdot 2^n \quad \forall\, n \geq n_0$$

Let $n_0 = 1$, $\Rightarrow 2^2 \leq c(2)$

$\Rightarrow 4 \leq c(2)$

$\Rightarrow \boxed{c=2}$ for $\boxed{n_0 = 1}$

<u>for, $n_0 = 2, c = 2$</u>

$2^{2+1} \leq 2(2^2)$

$2^3 \leq 2^3$

$8 \leq 8$ <u>true</u>.

<u>for $n_0 = 3, c = 2$</u>

$2^{3+1} \leq 2(2^3)$

$2^4 \leq 2 \cdot 2^3$

$16 \leq 16$ <u>true</u>.

$\therefore 2^{n+1} = O(2^n)$ is true and it holds for Big-Oh notation.

for $2^{2n} = O(2^n)$:

$$2^{2n} = O(2^n)$$

$$2^{2n} \leq c(2^n) \quad \forall \quad n > n_0$$

Let $n_0 = 1$ $\Rightarrow 2^2 \leq c(2^1)$

$\Rightarrow 4 \leq c(2)$

$\Rightarrow \boxed{c=2}$ for $\boxed{n_0 = 1}$

$\therefore$ for $n_0 = 2$, $c = 2$

$$2^{2 \cdot 2} \leq 2(2^2) \Rightarrow 2^4 \leq 2 \cdot 4$$

$$\Rightarrow 16 \leq 8 \text{ false.}$$

Thus, $2^{n} = O(2^n)$ does not holds true for Big-Oh notation.

This is because no constant is greater than all 2^n and so assumption leads to contradiction.

$$\therefore \begin{cases} 2^{n+1} = O(2^n) \\ 2^{2n} \neq O(2^n) \end{cases}$$

19/78065

6 (a) Write a program to find GCD of two numbers using recursion.

Ans:

```
#include <iostream>
using namespace std;
int GCD(int a, int b)
{
    if (b == 0)
        return a;
    return GCD(b, a % b);
}
int main()
{
    int a, b;
    cout << "\nEnter two positive integers : ";
    cin >> a >> b;
    cout << "--
> GCD of " << a << " and " << b << " is " << GCD(a, b) << endl
;
    return 0;
}
```

Output:

```
Enter two positive integers : 9 3
--> GCD of 9 and 3 is 3
```

(b) Write a program to represent graphs using adjacency matrices and check if it has Euler path.

Ans:

```
#include <iostream>
using namespace std;
int main()
{
    int **matrix, *deg;
    int v, sum = 0, count = 0;
    char ch;
    cout << "\nEnter the number of vertices : ";
```

```
    cin >> v;
    matrix = new int *[v];
    deg = new int[v];
    for (int i = 0; i < v; i++)
    {
        cout << endl;
        matrix[i] = new int[v];
        for (int j = 0; j < v; j++)
        {
            cout << "=> Is there a edge between " << i + 1 <<
" and " << j + 1 << " ? (y/n) : ";
            cin >> ch;
            if (ch == 'Y' || ch == 'y')
            {
                matrix[i][j] = 1;
            }
            else
                matrix[i][j] = 0;
        }
    }
    cout << "\n--> Given Adjacency Matrix : \n";
    for (int i = 0; i < v; i++)
    {
        cout << "\n\t";
        for (int j = 0; j < v; j++)
        {
            cout << matrix[i][j] << " ";
        }
    }
    for (int i = 0; i < v; i++)
    {
        sum = 0;
        for (int j = 0; j < v; j++)
        {
```

```
                sum += matrix[i][j];
            }
            deg[i] = sum;
        }
        cout << endl;
        for (int i = 0; i < v; i++)
        {
            cout << "\n--
> Degree of Vertex " << i + 1 << " : " << deg[i];
        }
        for (int i = 0; i < v; i++)
        {
            if ((deg[i] % 2) != 0)
            {
                count++;
            }
        }
        if (count == 2 || count == 0)
            cout << "\n\n--> Hence, it has an euler path.";
        else
            cout << "\n\n--> Hence, it has no euler path.";
        cout << endl;
        return 0;
    }
```

Output :

```
Enter the number of vertices : 3

=> Is there a edge between 1 and 1 ? (y/n) : n
=> Is there a edge between 1 and 2 ? (y/n) : y
=> Is there a edge between 1 and 3 ? (y/n) : y

=> Is there a edge between 2 and 1 ? (y/n) : y
=> Is there a edge between 2 and 2 ? (y/n) : n
=> Is there a edge between 2 and 3 ? (y/n) : y

=> Is there a edge between 3 and 1 ? (y/n) : y
```

```
=> Is there a edge between 3 and 2 ? (y/n) : y
=> Is there a edge between 3 and 3 ? (y/n) : n

--> Given Adjacency Matrix :

    0 1 1
    1 0 1
    1 1 0

--> Degree of Vertex 1 : 2
--> Degree of Vertex 2 : 2
--> Degree of Vertex 3 : 2

--> Hence, it has an euler path.
```

(c) Write a program to implement Insertion sort. Find the number of comparison during each pass and display the intermediate result.

Ans:

```
#include <iostream>
using namespace std;
void display(int ar[], int v)
{
    cout << " - [ ";
    for (int i = 0; i < v; i++)
    {
        cout << ar[i] << " ";
    }
    cout << "]";
}
int swap(int *ar, int a, int b)
{
    int temp = ar[a];
    ar[a] = ar[b];
    ar[b] = temp;

    return *ar;
```

```
}
int insertionSort(int *ar, int v)
{
    int i, j, key, totalCount = 0, count = 0;
    for (i = 1; i < v; i++)
    {
        key = ar[i];
        j = i - 1;
        count = 0;
        while (j >= 0 && key < ar[j])
        {
            ++count;
            swap(ar, j, j + 1);
            j = j - 1;
        }
        totalCount += count;
        display(ar, v);
        cout << " -> " << (count) << " comparisons" << endl;
        ar[j + 1] = key;
    }
    return totalCount;
}
int main()
{
    int *arr;
    int size, comparisons;
    cout << endl
         << endl
         << "Enter the size of array : ";
    cin >> size;

    cout << endl
         << "Enter the elements of the array : ";
    for (int i = 0; i < size; i++)
```

```
    {
        cin >> arr[i];
    }
    cout << endl;
    display(arr, size);
    cout << endl
         << endl
         << " - Sorting the array using Insertion Sort Algorit
hm...." << endl
         << endl;
    comparisons = insertionSort(arr, size);
    cout << endl
         << "--> Sorted. Here is the resulting array";
    display(arr, size);
    cout << endl
         << endl
         << "--> Total no. of comparisons : " << comparisons;
    return 0;
}
```

Output :

```
Enter the size of array : 5

Enter the elements of the array : 1 2 5 3 8

 - [ 1 2 5 3 8 ]

 - Sorting the array using Insertion Sort Algorithm....

 - [ 1 2 5 3 8 ] -> 0 comparisons
 - [ 1 2 5 3 8 ] -> 0 comparisons
 - [ 1 2 3 5 8 ] -> 1 comparisons
 - [ 1 2 3 5 8 ] -> 0 comparisons

--> Sorted. Here is the resulting array - [ 1 2 3 5 8 ]

--> Total no. of comparisons : 1
```