**Stendary decigonal element. **Algorithem ** **Algorit	(program to emplement sum of principle devigoried and
Algorithm Step 1: stoot Alip 2: a=0, sum=0 Step 4: 4 (m==n). // If balk yoto step 4.5 Print "Entir the co-spellant of matrix. for (i=0; i < n; i+t). for (i=0; i < n; i+t). Tread array [i][i] 3. 4.1. forent "In gourn matrix is output matrix element a[i][i] 4.2. for (i=0; i < m; i+t) { sum = sum + array [i][i] a=a+array [i][m-i-1] 3. 4.3. print sum of main diagonal Front sum 4.4. print sum of off diagonal Front a 4.5. elf.	
Step 2: a=0, scom=0 Step 3: Entir Eddregmedrick Tread m, n. Step 4: 4 (m==n). // Hy bedde goto step 4.5 Print Entir the co-efficient of matrix. bor (i=0; i < n; i+1). Tread array [i][i] 3. 4.1. promy "The gener matrix is output matrix is lemost a [i][i] 4.2. por (i=0; i < m; i+1) { sum = sum + array [i] [i] 3. 4.3. print scom of main chargened Print scom 4.4. print scom of of decigned point a 4.5. eds.	
Step 4: Enter order apmedicie Step 4: If (m==n). // If bedde goto step 4.5 When Enter the co-experient of matrix. bor (i=o; i < n; i+t). for (i=o; i < n; i+t). Tread array [i][i] 3. 4.1. promy "The gener matrix is. output matrix element a[i][i] 4.2. por (i=o; i < m; i+t) { *sum = sum + array [i][i] a = a + array [i][m-i-1] 3. 4.3. promy sum 4.4. primy sum 4.4. primy sum 4.5. else.	step 1: stoot
Step 4: if (m==n). // Hy balle goto step 4.5 prient "Entir the co-uplatent of matrix. bor (i=0; i < n; i+t). for (i=0; i < n; i+t). Tread array [i][i] 3. 4.1. prient "The getter matrix is. output matrix element a [i][] 4.2. por (i=0; i < m; i+t) from = sum + array [i][i] a=a+corray [i][m-i-i] 3. 4.3. prient sum of main diagonal prient sum u.4. prient sum of off diagonal point a 4.5. elst.	Slep 2: a=0, 8cm=0
step 4: If (m==n). // If bedde goto step 4.5 Prient "Entir the co-efficient of matrix. for (i=o; ; cn; i+t). for (j=o; ; cn; i+t). Tread array [i][i] 3. 4.1. forent "In genen matrix is. output matrix element a[i][i] 4.2. for (i=o; cm; i+t) { sum = sum + array [i][i] a=a+corray [i][m-i-l] 3. 4.3. prient sum of main diagonal prient sum 4.4. prient sum of off diagonal point a 4.5. elst.	stip 3: Enter ordor ormatrice.
prient "Entir the co-stituent of matrix. bor(i=0; i < n; i+1). f bor(i=0; i < n; i+1). 5 read array [i][i] 3. 4.1. provide "the getien matrix is output matrix element a[i][i] 4.2. por(i=0; i < m; i+1) { sum = sum + array [i][i] a=a+array [i][m-i-i] 3. 4.3. previous sum of materia diagonal brown sum 4.4. prient sum of off diagonal point a 4.5. else.	read m, n
prient "Entir the co-stituent of matrix. bor(i=0; i < n; i+1). f bor(i=0; i < n; i+1). 5 read array [i][i] 3. 4.1. provide "the getien matrix is output matrix element a[i][i] 4.2. por(i=0; i < m; i+1) { sum = sum + array [i][i] a=a+array [i][m-i-i] 3. 4.3. previous sum of materia diagonal brown sum 4.4. prient sum of off diagonal point a 4.5. else.	step 4: 4 (m==n). // Hy balle goto step 4.5
bor(1=0;) bor(1=0;) bor(1=0;) bor(1=0;) https://www.ciscons.com/lengths/ 4.1. print sum + array [i] [i] 4.2. print sum of main diagonal 4.3. print sum of main diagonal 4.3. print sum 4.4. print sum of off diagonal 4.5. elst.	
Freid wray [i][i] 3. 4.1. promly "In getter matrix is. output matrix element a[i][i] 4.2. polli=0; [cm; i+t] { sum = sum + array [i][i] a=a+array [i][m-i-1] 3. 4.3. prent sum of main diagonal prom sum 4.4. print sum of off diagonal print a 4.5. elst.	
Freid wray [i][i] 3. 4.1. promly "In getter matrix is. output matrix element a[i][i] 4.2. polli=0; [cm; i+t] { sum = sum + array [i][i] a=a+array [i][m-i-1] 3. 4.3. prent sum of main diagonal prom sum 4.4. print sum of off diagonal print a 4.5. elst.	$\{ bor(j=0;j< n;j+\epsilon) \}$
4.1. promy "The getter matrix is output matrix element a [i][i] 4.2. profi=0; i < m; i + t) { sum = sum + array [i][i] a=a+array [i][m-i-1] 3. 4.3. prem sum of main diagonal wind sum 4.4. prim sum of off diagonal point a 4.5. elst.	v
output medrois element a [i][] 4.2. fronti=0; i < m; i + t! { sum = sum + array [i][i] a = a + array [i][m-i-i] }. 4.3. prient sum of main diagonal print sum 4.4. prient sum of off diagonal print a 4.5. elst.	3
output medrois element a [i][] 4.2. fronti=0; i < m; i + t! { sum = sum + array [i][i] a = a + array [i][m-i-i] }. 4.3. prient sum of main diagonal print sum 4.4. prient sum of off diagonal print a 4.5. elst.	4.1. pront "The gelien matrix is
4.2. policojicmji+t) { sum = sum + adray [i] [i] a = a + adray [i] [m-i-1] }. 4.3. prem sum of maen diagonal brond sum 4.4. prem sum of off diagonal print a 4.5. elst.	
a=a+woray [i][m-i-i] 3. 4.3. previous sum of main diagonal provid sum 4.4. privid sum of off diagonal privit a 4.5. else.	
a=a+woray [i][m-i-i] 3. 4.3. previous sum of main diagonal provid sum 4.4. privid sum of off diagonal privit a 4.5. else.	{ Num = Num + after [] []]
3. 4.3. prem sum of main diagonal brown sum 4.4. print sum of off diagonal paint a 4.5. else.	3
4.3. prem sum of main diagonal word sum u.4. priend sum of off diagonal point a 4.5. else.	
word sum 4.4. print sum of off dicigonal print a 4.5. else.	3 .
word sum 4.4. print sum of off dicigonal print a 4.5. else.	4.3. prent sum of main diagonal
print a 4.5. else.	
print a 4.5. else.	4.4. priend sum of of dicigonal
from the given Irelia is not square matrice	4.5. else.
	from the given Irelia is not square matrice
	Stips. stop

flow chart. (stoot) a=0 Jam= 0 read m, n boll 影 (m==n) Town Brond Entir col Wount of matrice (i=o) icn; i+ Town ball (J=0; Scm; Trust words [1][Brand guin maliair atray [i][i] & balse Sum = fum + adday [[] [] a= a+ advay [i][m-i-1] Perent sum, a. power getter order is not square matrix