1- Representation

Sparse matrix is matrix in which data most are zero, So in result we wast space In memory and Processing time

Ex:-

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 0 & 0 & 1 \\ 2 & 1 & 0 & 0 & 7 \\ 3 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 8 \end{array} \quad 4 \times 4$$

we need to store only non-zero elements.

we have 2 methods

1- coordinate list $\xleftrightarrow{\text{same}}$ 3- column representation

2- comparsed Sparse row

let's perform coordinate List

we need 3 columns

Row	column	content
4	4	4
1	4	1
2	1	1
2	4	7
4	4	8

matrix dimensions ←

→ numbers of non-zero elements

now method 2 using 3 Arrays

Comparsed Sparse rows

non-zero elements ← $A[1, 1, 7, 8]$ In order!

$IA[\overset{0}{0}, \overset{1}{1}, \overset{2}{3}, \overset{3}{3}, \overset{4}{4}]$

عدد العناصر في الصف 0

عدد العناصر في الصف 1 + عدد العناصر في الصفوف السابقة

$JA[\underbrace{4, 1, 4, 4}]$

Columns of elements In Array A

Representation In Program

$$A = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 0 & 7 & 0 & 0 \\ 2 & 2 & 0 & 0 & 5 & 0 \\ 3 & 9 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 4 \end{array} \Rightarrow$$

4 x 5

	0	1	2	3	4	5
i	4	1	2	2	3	4
j	5	3	1	4	1	5
Value_ x	5	7	2	5	9	4

العمود الواحد

نسوي الجدول كامل

```
class Element
{

    int i;
    int j;
    int x ;
};
```

```
class Sparse
{
    int m; --> mAtrix dimension
    int n; -->
    int num; number of non-zero elements
  Element *e; To be ArrAy dynamic
}
```

Here is 2 row
colun

3- Program for Adding sparse matrix

هي نظرياً فقط إضافة الجدولين وننشئ جدول ثالث

video 160 for understanding concept of adding

$$S1 \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 0 & 3 & 0 & 0 \\ 2 & 4 & 0 & 0 & 0 & 7 \\ 3 & 0 & 0 & 5 & 0 & 8 \\ 4 & 0 & 6 & 0 & 0 & 0 \end{array}$$

S1 .m= 4
S1 . n = 5
S1 .num= 6

e Array is

0	1	2	3	4	5
1	2	2	3	3	4
3	1	5	3	5	2
3	4	7	5	8	6

$$S2 \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 0 & 0 & 0 & 2 \\ 2 & 0 & 5 & 0 & 0 & 6 \\ 3 & 4 & 0 & 8 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 9 \end{array}$$

S2. m=4
S2. n = 5
S2. num = 6

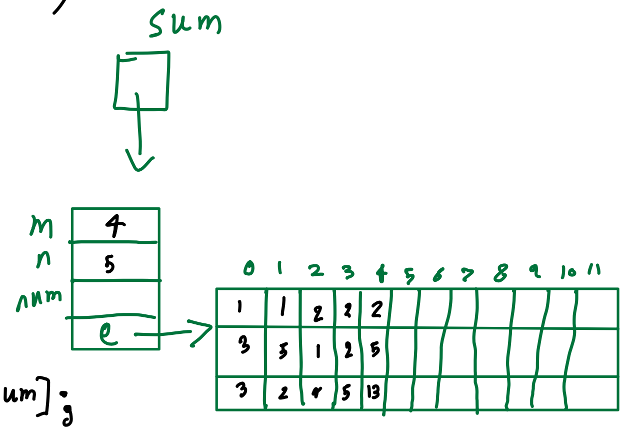
eArray is

0	1	2	3	4	5
1	2	2	3	3	4
5	2	5	1	3	5
2	5	6	4	8	a

we make function add

```
add (Sparse* s1, Sparse* s2)
 //first check can we add them?
  if(s1.m != s2.m || s1.n != s2.n)
          return 0;
  sparse *sum;
  sum= new Sparse;
  sum.n= s1.n;
  sum.m= s1.m;
 sum.e= new Element[s1.num+ s2.num];
 int i=j=k=0;

  while(i< s1.num && j<s2.num)
 if(s1.e[i].i< s2.e[j].i)
       Sum.e[k++] = s1.e[i++];
else if(s1.e[i].i > s2.e[j].i)
        sum.e[k++] = s2.e[j++];
else
{
   if(s1.e[i].j < s2.e[j].j)
       sum.e[k++] = s1.e[i++];
  else if(s1.e[i].j > s2.e[j].j)
       sum.e[k++] = s2.e[j++];
  else{
```



0	1	2	3	4	5	6	7	8	9	10	11
1	1	2	2	2							
3	3	1	2	5							
3	2	4	5	13							

```
      sum.e[k1]  =  s1.e[i++];
     sum.e[k++]+= s2.e[j++];
   }}
 num=k;
```

j