Vectors

- Lists can be used to represent mathematical objects called vectors & matrices
- Representing Vectors:
  - A vector is a mathematical object consisting of a sequence of numbers called the components of a vector
  - Row Vector (1 x m) and Column Vector (m x 1)
  - An n-dimension vector can be represented as a one-dimensional list in Python with n elements present in the array
    - X=(1, 2, 4) → X[0]=1, X[1]=2, X[2]=4
  - Two vectors can be multiplied together forming the Scalar Product which is the sum of the products of corresponding components
    - If X = (1,2,4) and W=(2, 3, 1)

Their scalar product is X * W = 1*2 + 2*3 + 4*1 = 12

Vectors

- In general: Scalar product of two vectors of dimension n is:

$$\sum_{k=1}^{n} X_k * W_k$$

Where X_k and W_k represent the kth components of X and W respectively

- Coding the scalar product computation of two such vectors involves traversing the vectors component by component:

```
sum_prod = 0
for k in range(n):
    sum_prod += x[k] * w[k]
```

Matrices

- Representing Matrices:
  - A matrix is a mathematical object consisting of a rectangular arrangement of numbers called the elements of the matrix
  - An ***m*** x **n** matrix consists of m rows and n columns, each row is an n-dimensional vector and each column is an m-dimensional vector
  - An ***m*** x **n** matrix can be implemented in Python as a two-dimensional array with m rows and n columns
  - The element in the ith row and jth column is denoted by a_{ij} which is a[i][j] noting that the starting point for i and j is 0

$$A = \begin{matrix} 1 & 1 & 1 \\ 2 & 3 & 1 \\ 1 & -1 & -1 \\ 0 & 1 & 2 \end{matrix}$$

➔ represent this as a 4 x 3 list

Matrices

- Scalar Multiple of a Matrix
  - Given any matrix A and any number c (a *number* is sometimes referred to as a *scalar*), matrix cA is obtained from the matrix A by multiplying each element of A by c
  - A and 3A
  - If c=-1, A is sometimes written as –A
- Addition of Two Matrices
  - If A and B are two matrices of the same order (say, m x n)
    - C = A + B is defined to be the m x n matrix whose *ij*th element is $a_{ij} + b_{ij}$

Vectors and Matrices

- Multiplying a Matrix by a Vector
  - If A is an ***m*** x **n** matrix and X is an n-dimensional vector, we can form the product of A and X denoted by A * X, to yield an m-dimensional vector V
  - A matrix with m rows and n columns can be multiplied on the right only by a vector of dimension n (and can be multiplied on the left only by a vector of dimension m)
  - To yield the result vector, the ith component of V, v[i] is the scalar product of the ith row of the matrix A and vector X Consider a 3-dimensional vector X (1, 2, 2) and the 4x3 matrix from our example:
    - Multiply the matrix by the vector to get a result Vector V:
    - V[1] = (2, 3, 1) * (1, 2, 2) = 2*1 + 3*2 + 1*2 = 10

Vectors and Matrices

– Mathematical formula:

$$V_i = \sum_{k=1}^{n} A_{ik} * X_k$$

– Translating into Python, to compute the Vith term:

```
v[i] = 0
for k in range(n):
    v[i] += a[i][k] * x[k]
```

(The code would have to be completed to compute the entire result vector)

Matrix Multiplication

- Matrix Multiplication
  - A vector can be viewed as a special case of a matrix, for example a n-dimensional vector can be considered to have 1 row and n columns (or n rows and 1 column)
  - Two matrices A and B can be multiplied together yielding a new matrix, C = A * B if the number of columns in A is equal to the number of rows in B
  - If A is an ***m*** x **n** matrix and B is ***n*** x ***p***, then the product matrix C will be ***m*** x ***p***
  - Element C_{ij} is the scalar product of the ith row of A and jth column of B (i.e. vector multiplication of the ith row of A and the jth column of B)
  - We would need to have embed the computation of C_{ij} in an appropriate looping mechanism like:

```
for i in range(m):
    for j in range(p):
        ….compute c[i][j] ….
```