$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

A

$$\begin{bmatrix} 3 & 5 \\ 6 & 7 \\ 8 & 9 \end{bmatrix}$$

B

~~A * B~~

Numbers of columns of A = numbers of rows of B

$$\begin{bmatrix} 2 & 3 & 6 & 7 \\ 8 & 9 & 4 & 1 \end{bmatrix}$$

C

A * C ✓

C * A ✗

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

A

$$\begin{bmatrix} 2 & 3 & 6 & 7 \\ 8 & 9 & 4 & 1 \end{bmatrix}$$

C

$$= \begin{bmatrix} \textcircled{} & 0 \\ & \\ & \end{bmatrix}$$

3

number of rows of first matrix

4

number of columns of the second matrix

$v1 = (x_1, x_2, x_3, \cdots, x_d)$

$v2 = (x_1', x_2', x_3', \cdots, x_d')$

$$(x_1 \cdot x_1') + (x_2' \cdot x_2') + \cdots + (x_d \cdot x_d')$$

$$= \sum_{i=1}^{d} (x_i, x_i')$$

```
sum = 0;
for(int i=0; i<d; ++i)
{
    sum += (x[i] * x'[i]);
}
```

A A_r , A_c

B B_r , B_c

$C = A * B$. $\boxed{A_c = B_r}$

C has A_r rows & B_c columns

$C \in \mathbb{R}^{A_r \times B_c}$

```
for (int i=0; i<Ar; i++)
    for(int j=0; j<Bc; j++)
    {  // ith row dot product with jth column
       C[i][j]=0;
       for(int k=0; k<Ac; k++)
          c[i][j] += (A[i][k] * B[k][j])
    }
```

$A[i][k]$: ith row of A

$B[k][j]$: jth column of B

```
void to_str(char** s, int d)
{
    *s = new char[10];
    sprintf(*s, "%d", d);
}

int main()
{
    char * s;
    to_str(&s, 23);
    delete [] s;
}
```