

1. Write a function that calculates the product of a matrix and a vector. Let us denote the matrix as

$$A = (a_{ij})_{3 \times 3} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

and the vector

$$X = (x_i)_3 = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} .$$

The product of the matrix and the vector $AX = Y = (y_i)_3$ is given by

$$y_i = \sum_{j=1}^3 a_{ij} x_j , \text{ that is}$$

$$AX = Y = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 \end{pmatrix} .$$

The prototype of the function could be

```
void mvmult(double a[][3], double x[], double y[]);
```

where a is the matrix, x the vector and y a vector where the result is to be stored.

2. Write a function that calculates how many times a given character appears in a string. The prototype could be

```
int ccount(char str[], char c);
```

where str is the string, c a character. The function returns how many times the letter appears in the string. Hint: write a loop that looks for the end character of the string, '\0'.

3. Write a function that calculates lottery numbers. You'll need a function that gives you random numbers. Look for the function rand() in the standard library, and see how it works. Then make a program that calculates seven random numbers between 1 and 39. When written like this, any given number may appear several times in the numbers. Try changing your program so that it will give any number only once!
4. Extra: Write an anagram tester. The function should return true if the two strings given as arguments are anagrams of each other.


```
int is_anag(char stra[], char strb[]);
```