The k-means algorithm is a popular method to cluster data, i.e., find unknown groups in the data. In its simplest form it consists of the following steps:

1. Draw K random data points as initial centers.
2. Assign each data point to the cluster corresponding to the closest center.
3. Compute new centers for each cluster: minimum average distance to points in cluster. For Euclidean distance this corresponds to the column-wise means of all data points in the cluster.
4. Repeat from step 2 until nothing changes.

Exercise 1
Write a function which calculates the Euclidean (=squared) distances between all pairs of rows of two matrices x and y.

Exercise 2
Implementient the k-means algorithm, use function `which.min()` to determine the clusters of each point.

Exercise 3
Use Rprof() to determine which parts of your k-means implementation are fast or slow.

Exercise 4
Define class `Kmeans` for the result and write a `print()` method for it.

Exercise 5
Write a `plot()` method for the class: a scatterplot matrix with different colors for the clusters and optionally cluster centers using different symbols.

Exercise 6
Write S4 versions for the above.

Exercise 7
Create a new generic `Kmeans4` which has methods for

1. numeric data matrices (what we did until now)
2. data frames: convert to a numeric matrix using rules like: accept columns with numeric values, unordered factors with a maximum of two levels (convert to 0/1), and ordered factors (convert to integer).