1. Create a function that ask a number to the user. The function must show the elements of the vector until that the element is found in the vector. A=c(2,1,4,3,5)

2. Create a function that show the elements until finding two equal elements in the same vector. A=c(1,2,3,6,5,4,4,6)

3. Create a function. Then a number and a numeric matrix are passed by parameter. The function has to show the elements of the matrix; if the passed number is in the matrix, the function won't continue to show more elements. (Do the exercise with the different loops)

   A) Go through the matrix by row

   B) Go through the matrix by column

4. Create a function. Then a number and a numeric matrix are passed by parameter. The function has to check if the number is in the matrix and count the number of times that it appears in the matrix. (Do the exercise with different loops)

5. Perform a function that defines two matrices, matrix A and B; check if there is any common number, if so, replace it with the value -1.

   $A = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$ $B = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$

6. Create a function called "sumMatrix". The function has to sum the matrix by row and insert the result into a vector called "RowSum"; and the sum by columns and it will insert in a vector that is called "SumColumns"

7. Create a function called "countNumMatriz". The function must count how many times each number of matrix appears in the matrix; the result must be inserted in another matrix, called B, both the number and the quantity of that number appears

   $A = \begin{pmatrix} 1 & 3 \\ 3 & 4 \end{pmatrix}$ $B = \begin{pmatrix} 1 & 1 \\ 3 & 2 \\ 4 & 1 \end{pmatrix}$

8. Define a function that creates a vector A, for example, A = (2,4,5,7); you create another vector B, for example B = (1,2,3,6,8). The function must create two resulting vectors, one vector that returns the matching numbers and in another one the non-matching numbers. Both vectors must be ordered from least to greatest. There should be no repeated numbers in the vectors. Res1 = (2); Res = (1,3,4,5,6,7,8)

9. Define a function that creates a vector A, for example, A = (2,4,5,7); you create another vector B, for example B = (1,2,3,6,8). The function must create two resulting vectors, one

vector that returns the even numbers and another the odd numbers. Both vectors must be ordered from highest to lowest. If there are matching numbers, it will only appear once in the resulting vector. Res1 = (8,6,4,2); Res = (7,5,3,1)

10. Define a function that creates a vector A ordered from least to greatest, for example, A =(2,4,5,7); ask for a number by keyboard that is between the minimum number of the vector and the maximum; in the example it would be 2 and 7; check if the number keyboard input exists in vector; if it does not exist you must insert it in the correct position, in order from least to greatest:

    a) using the sort ().

    b) without using the sort ().

11. Create a function called "CalculateHeightandWeights". The function should create a matrix with the data shown in the table including the row names and column ones. The function should:

    - Calculate what is the height and the minimum weight.

    - Calculate how many people have exceeded the average weight and height of the table data

    - Request height and weight to the user (consider that it is valid data). It must count how many people qualify.

    a) Perform the exercise with while loops
    b) Perform the exercise with for loops
    c) Perform the exercise with repeat loop

	Height	Weight
Pepe	1.70	70
John	1.65	70
Antonio	1.80	76

12. Create a function called "CalculateData". The function must create a matrix with the notes of the students of two subjects. The function should display the data as shown in the example, including the names of the rows and columns. The function should:

    - Calculate which is the minimum and maximum grade of each subject

- Calculate which is the minimum and maximum grade of both subjects

- Calculate how many students have passed both subjects

- Ask the teacher for a grade (consider that the grade is valid); You must count how many students have obtained that grade.

	Programming	Math
Gema	8	9
Rosa	3	8
Lucia	2	6

a) Perform the exercise with while loop

b) Perform the exercise with for loop

c) Perform the exercise with repeat loop

13. Define a function called "StudentData" that defines a data structure that allows describing Table1.Insert the values that this table contains, name of the columns and rows, as it appears in Table1. In the same function, insert a column with the sum of the note of Partial 1 and Partial 2 of the students. The resulting table should be like table 2.

	Partial1	Partial2
Ana	2.5	2.5
Pepe	2	2
Nacho	0.3	1.6
Bea	2.5	0.9
Gema	2.1	2.4
Alba	1.2	2.8

Table 1. Partial 1 and Partial 2

	Partial1	Partial2	Grades
Ana	2.5	2.5	5
Pepe	2	2	4
Nacho	0.3	1.6	1.9
Bea	2.5	0.9	3.4
Gema	2.1	2.4	4.5
Alba	1.2	2.8	4

Table 2. Partial 1, Partial 2 and Grades

In the same function, insert two more columns. A column with the data in which it appears if each student has delivered an extra work. In the second column the final grade must be calculated. If the student has submitted the work, extra point will be added. In addition, you must insert a new student (Javier) who was not there by mistake. The final table is as it appears in Table 3

	Partial1	Partial2	Grades	Submit	Final Grade
Ana	2.5	2.5	5	YES	6
Pepe	2	2	4	YES	5
Nacho	0.3	1.6	1.9	NO	1.9
Bea	2.5	0.9	3.4	YES	4.4
Gema	2.1	2.4	4.5	YES	5.5
Alba	1.2	2.8	4	YES	5
Javier	2.3	2.1	4.4	YES	5.4

Tabla3. Partial1, Partial2, Grades, Submit and Final Grades

In the same function, calculate the number of students who have passed continuous assessment and how many have failed it.