

GUÍA DE EJERCICIOS

Nº 3

ING. LUISINA DE PAULA

**DESARROLLO
WEB FULLSTACK
CON JAVA**

Guía de Ejercicios Nº 3: Vectores y Matrices

1) Vectores

Resolver los siguientes ejercicios. Intentar ejecutarlos y probarlos con datos de prueba.

- Llevar a cabo un programa que incluya un vector que almacene los nombres: Alejandra, Leonardo, Rosa, Guillermo, Gabriel, Daniel, Luisa y Ludmila. Realizar un recorrido del vector para cargar los datos y otro recorrido para mostrar los mismos por pantalla.
- Realizar un programa que permita la carga de 10 números en un vector. Una vez cargados, se necesita que el programa determine cual es el mayor y cuál es el menor de ellos.

Resolución de búsqueda del mayor (en Pseudocódigo): <https://youtu.be/cxi8dH2VJYE>

- Realizar un programa que permita la carga de 15 números en un vector. Una vez cargados, se necesita que el programa cuente e informe por pantalla cuántas veces se cargó el número 3.

Resolución de búsqueda del mayor (en Java): <https://youtu.be/C5u72RuvrY4>

- Un empleado desea almacenar en un vector sus 12 sueldos del año. A partir de esto, necesita un programa que determine y muestre por pantalla la suma total de sus 12 sueldos y el promedio de sueldo que tuvo en el año.
- En **tres vectores diferentes** se guardan los nombres, temperaturas mínimas y máximas de 5 ciudades de la provincia de Misiones. En el primer vector se guardan los nombres de las ciudades, en el segundo las temperaturas mínimas alcanzadas y en el tercero las temperaturas máximas alcanzadas en la última semana. Se necesita un programa que permita la carga de las ciudades, sus temperaturas mínimas y máximas; además, deberá poder informar por pantalla cual fue la ciudad con la temperatura más baja y cual con la temperatura más alta (dando a conocer al mismo tiempo la cantidad de grados).

| |
|------------|
| Oberá |
| Posadas |
| L. N. Alem |
| ... |

| |
|-----|
| 16 |
| 18 |
| 15 |
| ... |

| |
|-----|
| 29 |
| 32 |
| 28 |
| ... |

2) Matrices

Resolver los siguientes ejercicios. Intentar ejecutarlos y probarlos con datos de prueba.

- Llevar a cabo un programa que permite cargar completamente con números 5 una matriz de 4x5 (4 filas, 5 columnas).

- b. Llevar a cabo un programa que permita completar con números 1 toda la diagonal principal de una matriz de 6 x 6. Entiéndase por diagonal principal aquella que comienza en [0,0] y termina en [6,6].
- c. Llevar a cabo un programa que permita cargar mediante teclado una matriz de 3 x 3 (3 filas, 3 columnas) con razas de perros. Una vez cargada, se debe permitir al usuario ingresar por teclado una raza de perro que quiera buscar; el programa deberá poder encontrarla y mostrar un mensaje con la posición (fila, columna) donde la encontró. En caso de que la raza de perro no se encuentre en la matriz, se deberá informar la situación mediante un mensaje por pantalla.
- d. En una tabla de 4 filas y 4 columnas se guardan las notas de 4 alumnos de secundario. Cada fila corresponde a las notas y al promedio de cada alumno.
 - i. Se necesita un programa que permita a un profesor cargar en las 3 primeras posiciones de cada fila las notas del alumno y que en la última columna se calculen los promedios.
 - ii. Una vez realizados los cálculos, se desea mostrar las 3 notas de cada alumno y el promedio correspondiente recorriendo la matriz. Ejemplo:

| | Nota 1 | Nota 2 | Nota 3 | Prom. |
|----------|--------|--------|--------|-------|
| Alumno 1 | 8 | 8 | 7 | ? |
| Alumno 2 | 7 | 9 | 10 | ? |
| Alumno 3 | 10 | 9 | 5 | ? |
| Alumno 4 | 4 | 9 | 8,50 | ? |

Resolución de búsqueda del mayor (en Java): <https://youtu.be/QnXOZkPfxG8>

- e. Una compañía de vuelos cuenta con 6 destinos a los que realiza 3 vuelos diariamente, uno por la mañana, otro al mediodía y otro por la noche. Para administrar estos datos, utiliza una matriz, donde cada fila es un destino y en cada columna se guarda la cantidad de asientos disponibles. Por Ejemplo:

| | | 0 | 1 | 2 |
|---|----------------|--------|----------|-------|
| | | Mañana | Mediodía | Noche |
| 0 | Rio de Janeiro | 120 | 15 | 30 |
| 1 | Cancún | 65 | 1 | 2 |
| 2 | Madrid | 4 | 52 | 7 |
| 3 | Roma | 46 | 32 | 16 |
| 4 | Milán | 64 | 23 | 11 |
| 5 | Iguazú | 61 | 12 | 91 |

- i. Se necesita un programa que permita la carga de la matriz con la cantidad de asientos para cada vuelo.
- ii. Al mismo tiempo, el programa debe permitir a un usuario ingresar el número de destino al que quiere dirigirse, el número de vuelo (dependiendo si quiere viajar a la mañana, al mediodía o a la noche) y la cantidad de pasajes que necesita.
- iii. A partir de la solicitud del usuario, el programa debe controlar si hay la cantidad suficiente de asientos para la cantidad de pasajes que se requiere. En caso de que así sea, se debe mostrar un cartel por pantalla que diga “su reserva fue realizada con éxito” y se debe descontar del total de asientos los solicitados por el usuario. En caso de no haber más asientos disponibles, se debe informar otro cartel que diga “disculpe, no se pudo completar su operación dado que no hay asientos disponibles”.

Desde la compañía de vuelos manifiestan que NO CONOCEN cuantas ventas/reservas se hacen por día. Por lo cual, para finalizar las ventas se ingresa la palabra “finish”.