

MyOpenLab

Versión 3.015

Manejo de datos en Matrices y Tablas

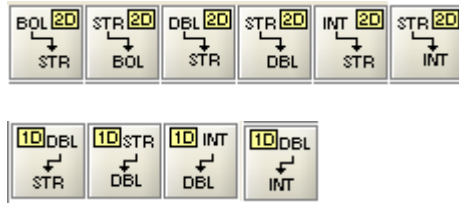
<http://myopenlab.de>
<http://es.myopenlab.de>



Prof. José Manuel Ruiz Gutiérrez
jruiz1@platea.pntic.mec.es

Índice

1. Tratamiento de Matrices y Tablas en MyOpenLab
2. Ejemplos Básicos de manipulación de datos en matrices.
 - 2.1. Conversión de una matriz 2D de datos tipo decimal (DBL) en una matriz de datos tipo string (STR)
 - 2.2. Lectura de datos de una tabla 2D
 - 2.3. Escribir datos en una tabla
 - 2.4. Suma de dos matrices del mismo rango
 - 2.5. Producto de dos matrices del mismo rango
 - 2.6. Convertir tipos de datos de una matriz 2D
 - 2.7. Cargar una tabla desde un fichero en formato csv y mostrarla en pantalla,
 - 2.8. Cargar un fichero CSV que contiene una tabla, mostrarla, llevarla a memoria y mostrarla desde memoria
 - 2.9. Lee un fichero CSV lo muestra en pantalla (tabla) y después lo graba en otro fichero CSV
 - 2.10. Escribiendo datos en una matriz



b) *Conversión de matrices 1D a 2D y viceversa*



c) *Funciones de entrada de datos para matrices 1D y 2D.*



d) *Operaciones con matrices*



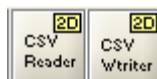
e) *Otras operaciones con matrices*



f) *Leer y escribir datos en una matriz 1D o 2D y llevar una matriz a memoria*



g) *Leer o escribir un fichero tipo CSV conteniendo una matriz 2D*



2. Ejemplos Básicos de manipulación de datos en matrices.

A continuación vamos a exponer algunos ejemplos básicos en los que intervienen datos en forma de matriz.

1.1. Conversión de una matriz 2D de datos tipo decimal (DBL) en una matriz de datos tipo string (STR)

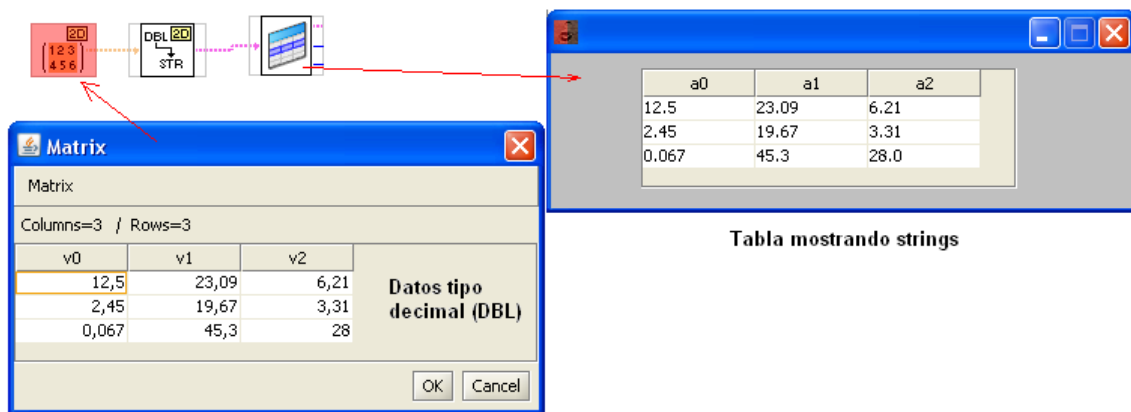


Figura 2

En la figura 2 vemos un bloque de función de carga de datos, matriz 2D de 3x3, que se pasa a un bloque de conversión y luego se sacan los datos convertidos en un bloque de visualización de tablas.

1.2. Lectura de datos de una tabla 2D

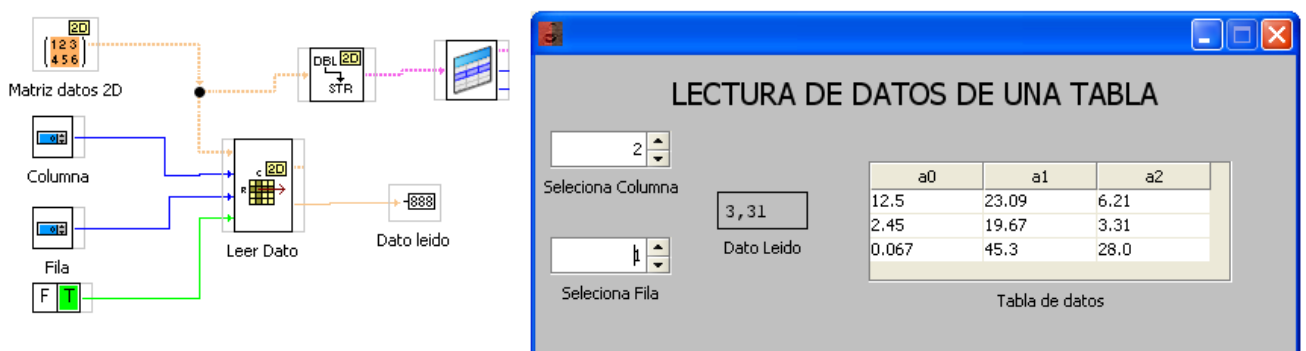
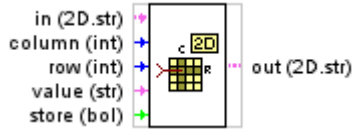


Figura 3

En la figura 3 vemos que los datos de la matriz 2D son leídos mediante el bloque "Leer dato" que nos permite seleccionar la fila y columna mediante sus correspondientes entradas y obtenemos en su salida el dato leído.

1.3. Escribir datos en una tabla

En este ejemplo vamos a ver la manera de escribir datos en una tabla. Para realizar la

aplicación recurrimos al bloque de función  en el que podemos seleccionar la fila y columna mediante las entradas de tipo “int” “column” y “row” y la entrada “value” por donde colocamos el dato que queremos escribir en la celda seleccionada. Bastará dar la orden “store” para almacenar el dato en esa posición.

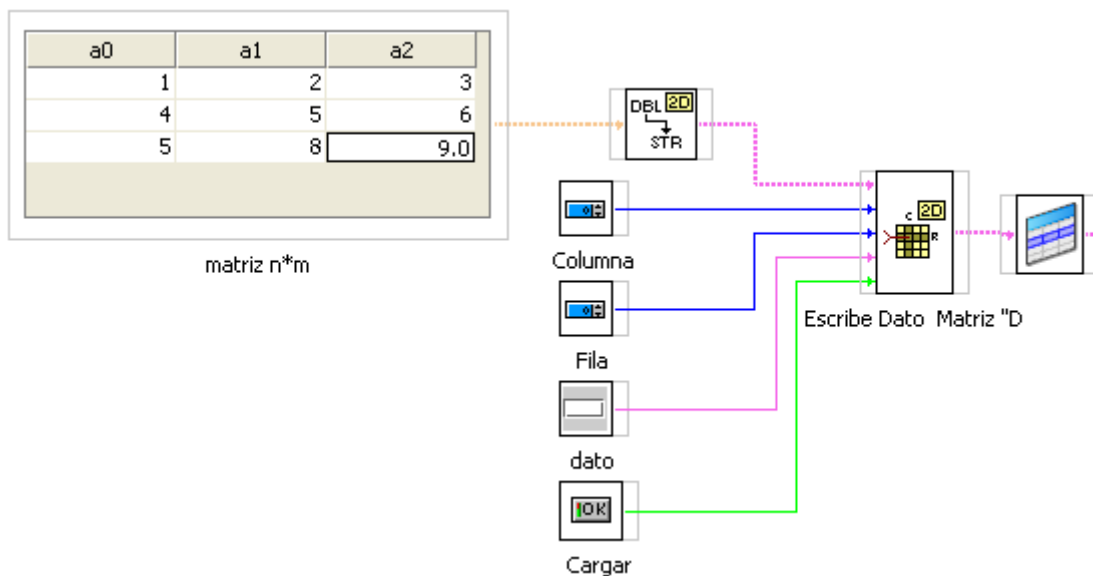


Figura 4

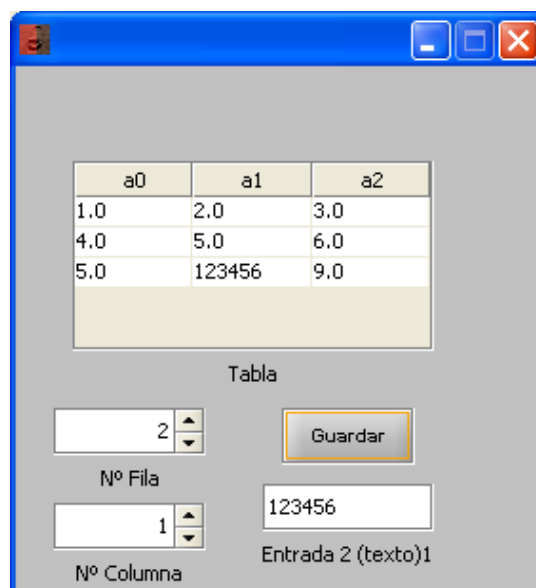


Figura 5

En la figura 5 vemos el aspecto de la ventana de simulación. Se puede observar como hemos escrito el dato “123456” en la fila2, columna1.

1.4. Suma de dos matrices del mismo rango

El ejemplo de la figura 6 nos muestra como operar para sumar los datos de dos matrices y mostrar los resultados.

Los datos de las matrices a y b se llevan a un elemento convertidor de datos que los convierte en datos tipo STR (esto es necesario porque el visor solo admite datos tipo STR). Por otra parte estos datos de las matrices a y b se suman mediante el correspondiente bloque funcional de suma y se saca el resultado a un visor.

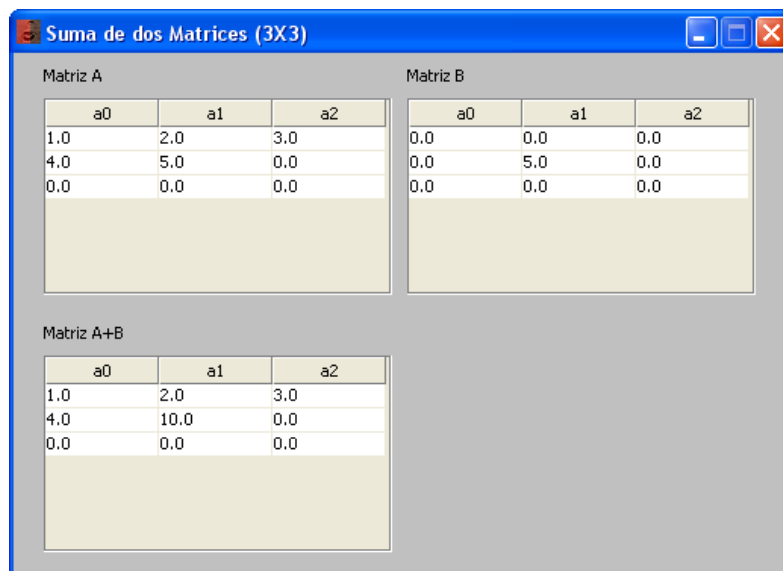
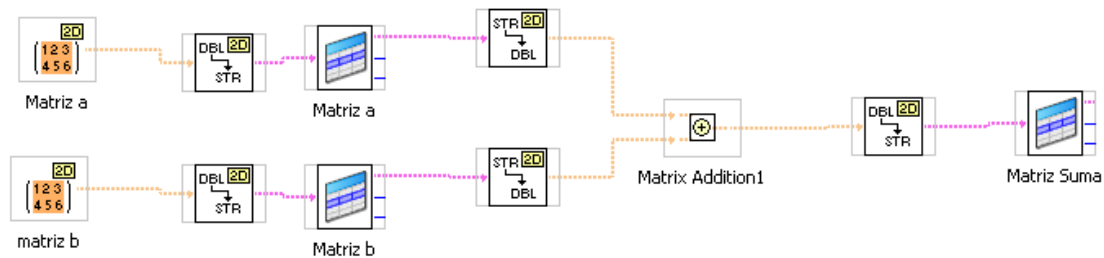


Figura 6

En la figura 6 vemos el aspecto de la matriz suma "Matriz A+B".

Los datos de las matrices A y B se pueden modificar en las tablas de la pantalla de visualización siempre que estas no estén configuradas en modo solo lectura (propiedad

`readOnly` sin habilitar.

1.5.Producto de dos matrices del mismo rango

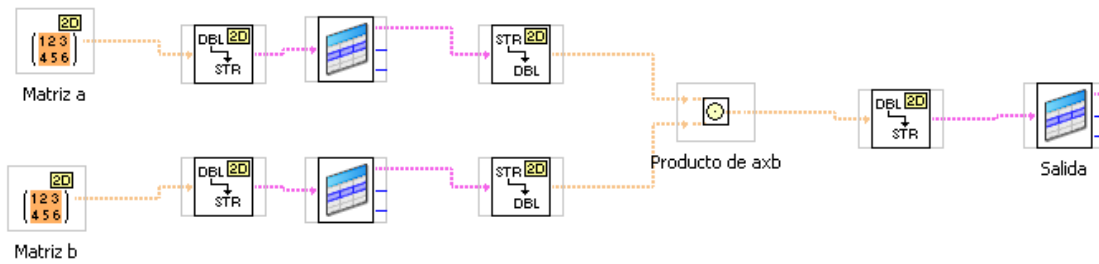


Figura 7

Al igual que en el anterior ejemplo los bloques de función están dispuestos de la misma forma con la salvedad de que en este caso la operación con las matrices es el producto.

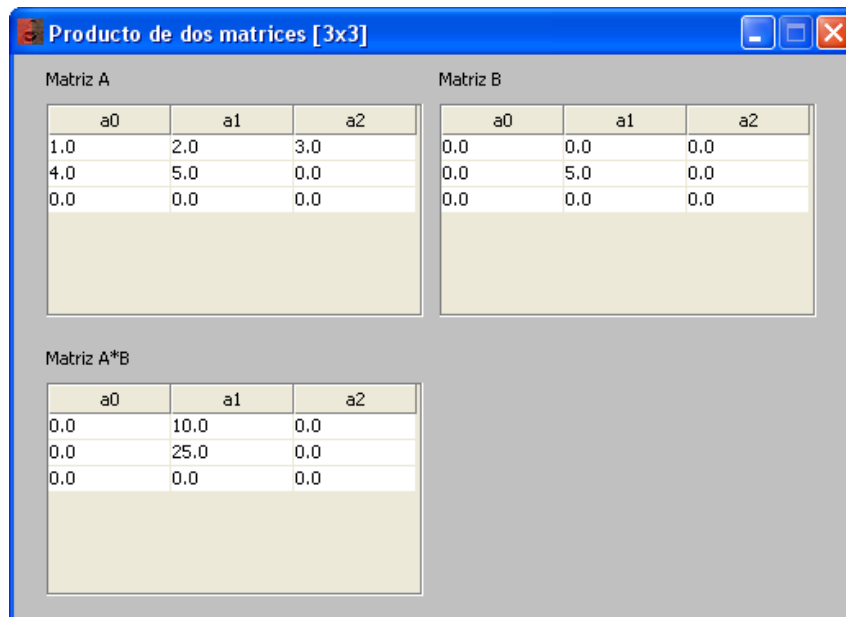


Figura 8

1.6. Convertir tipos de datos de una matriz 2D

En este ejemplo se trata de calcular el valor de la función seno y recoger la tabla generada por el bloque “Genera Gráfico1” pasarla a una matriz de tipo 2D convertirlas en una matriz de tipo STR y sacarla por un visor. Se han puesto dobles conversiones de matrices para ver como se realiza esta conversión sin ningún inconveniente.

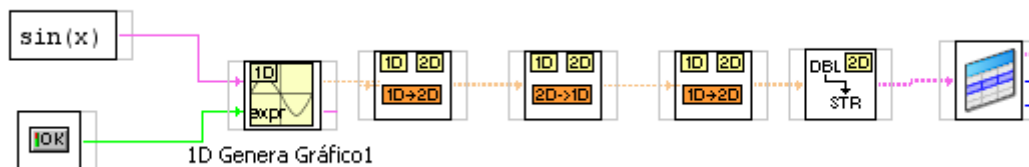
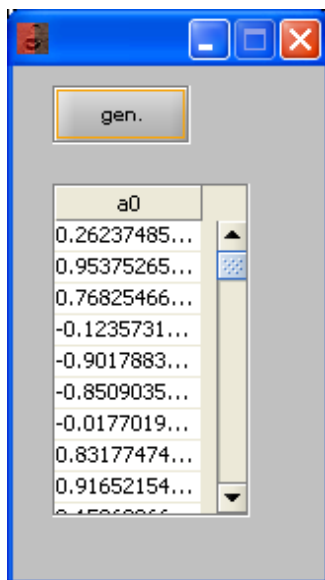


Figura 9



En la figura 11 se ha realizado una variación sobre el mismo modelo. En este caso se saca la tabla a través de la herramienta de visualización “Lista” que lo único que hace es mostrar una tabla de tipo 1D, es decir en este caso basta con recoger la tabla de salida del generador de función convertir el dato DBL en STR y mostrarlo

Figura 10

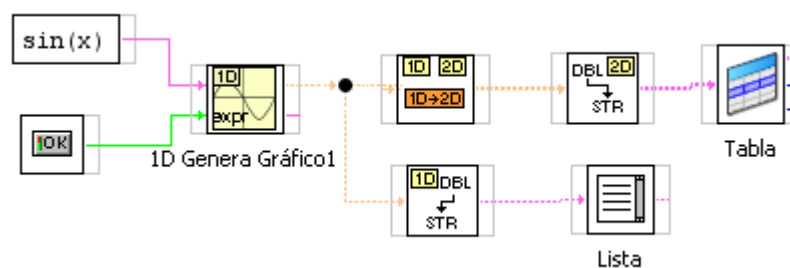


Figura 11

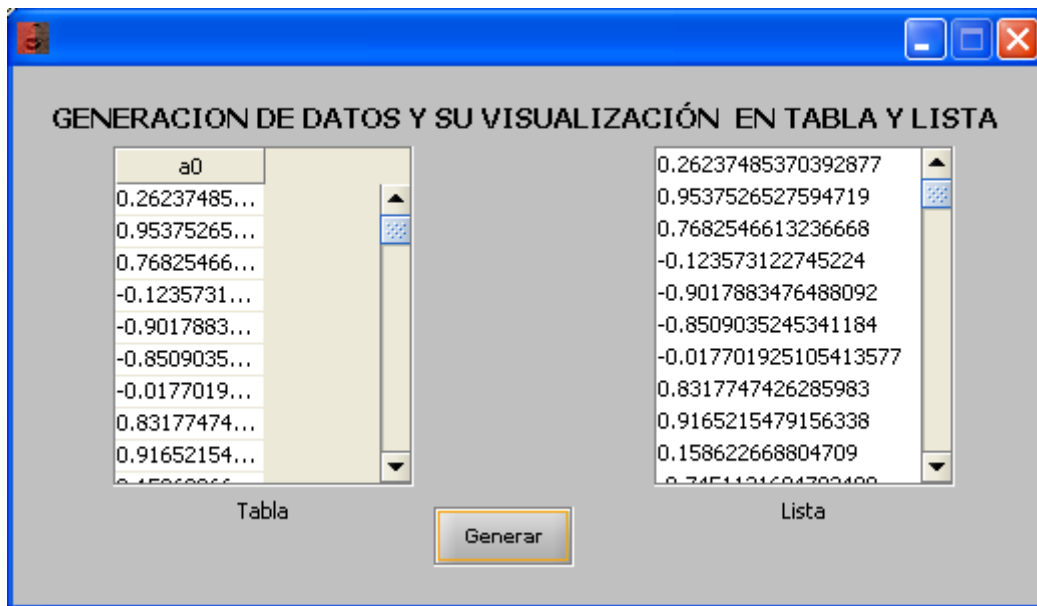


Figura 12

La figura 12 nos muestra la ventana de simulación de la aplicación

1.7. Cargar una tabla desde un fichero en formato csv y mostrarla en pantalla.

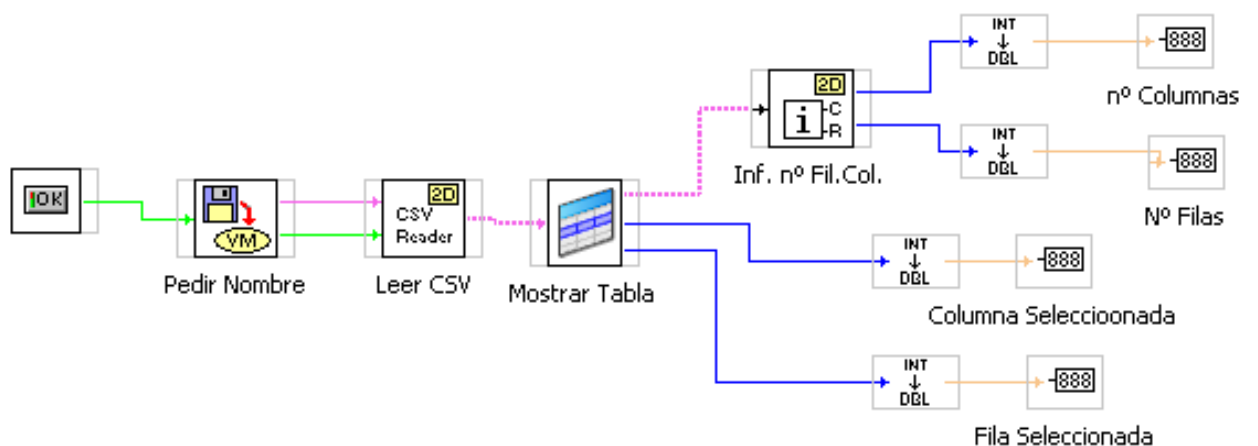


Figura 13

Con este ejemplo queremos dejar claro como es posible leer un fichero tipo CSV y poner sus datos en el formato de tabla.

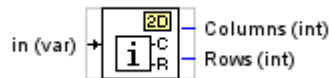
En la figura vemos que el bloque “Pedir Nombre” se encarga de mostrar una ventana de selección de fichero desde la que podemos seleccionar el tipo de fichero *.CSV que queramos leer.

El bloque “Leer CSV” se encarga de leer realmente el fichero y lo entrega al bloque visor de tabla “Mostrar Tabla”.

El bloque visor de tabla presenta en su salida los datos de la matriz 2D que lee en su



entrada y además una valiosa información que hace referencia al número de columna y de fila correspondientes a la celda que tengamos marcada con el ratón cuando estemos en modo simulación..



Del mismo modo el bloque nos permite averiguar el número total de filas y de columnas de la tabla que coloquemos en su entrada.

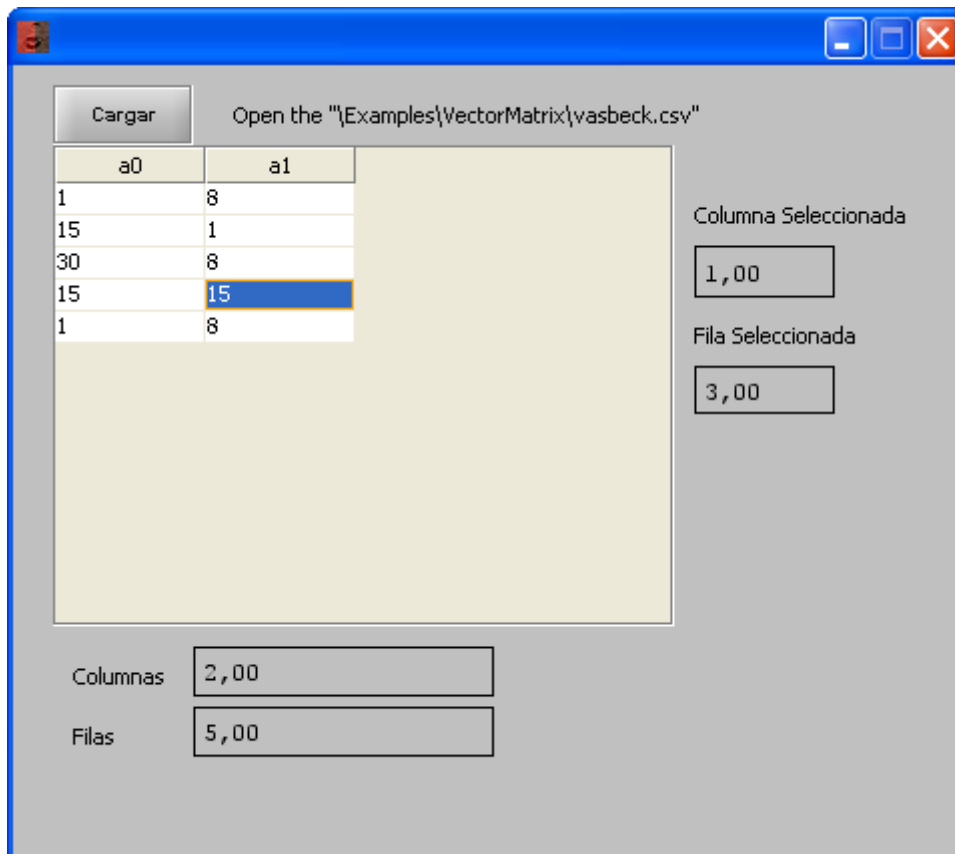


Figura 14

1.8. Cargar un fichero CSV que contiene una tabla, mostrarla, llevarla a memoria y mostrarla desde memoria

En el siguiente ejemplo se trata de realizar la carga de los datos incluidos en un fichero tipo SCV, mostrarlos en una tabla y a la vez llevarlos a memoria y después sacarlos también en una tabla.

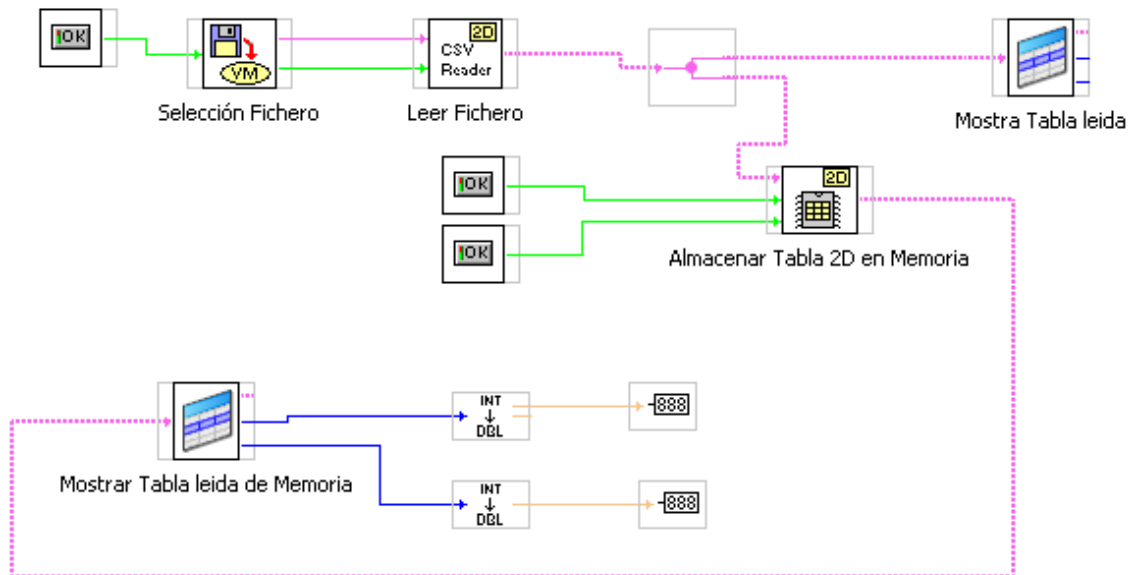


Figura 15

Esta opción de poder leer ficheros CSV (compatibles con EXCEL) es muy útil porque permitirá a MyOpenlab realizar cargas y descargas de datos sobre ficheros para poder realizar el tratamiento de los datos que mejor convenga.

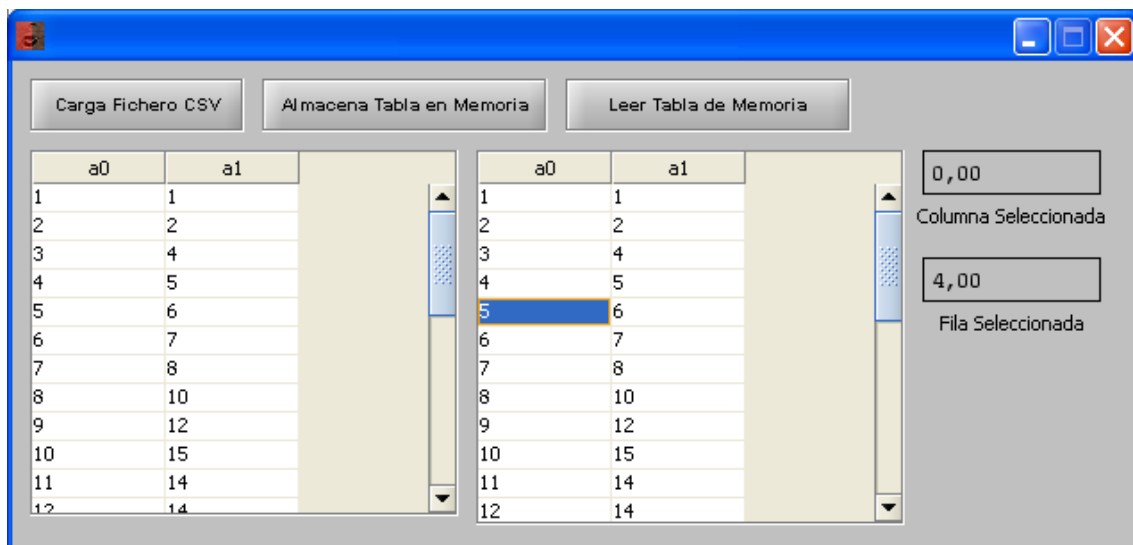


Figura 16

1.9. Lee un fichero CSV lo muestra en pantalla (tabla) y después lo graba en otro fichero CSV

Con este ejemplo se trata de ver la posibilidad de grabar una tabla que previamente hemos leído con los bloques “Selecciona fichero” seleccionamos el fichero que queramos leer o que queramos escribir según sea el caso.

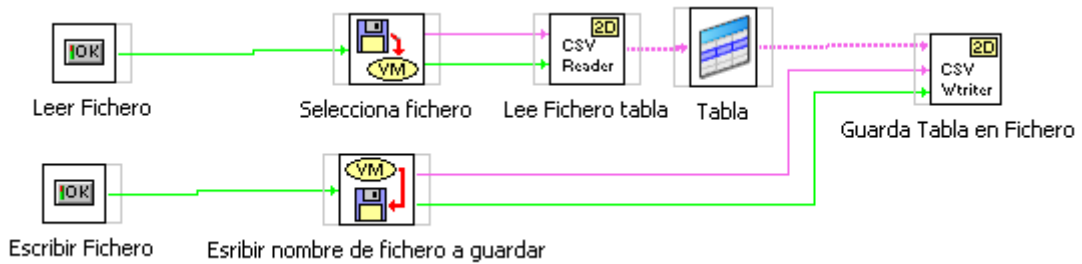


Figura 17

La función “Leer fichero tabla” se encarga de leer el fichero y convertirlo en una tabla tipo 2D. Por otro lado el bloque “Guarda Tabla en Fichero” se encarga de guardar la tabla 2D leída en un fichero.

Con el botón “Leer fichero” damos la orden para abrir y leer el fichero. Con el botón “Escribir Fichero” damos la orden de guardar la tabla leída en un nuevo fichero.

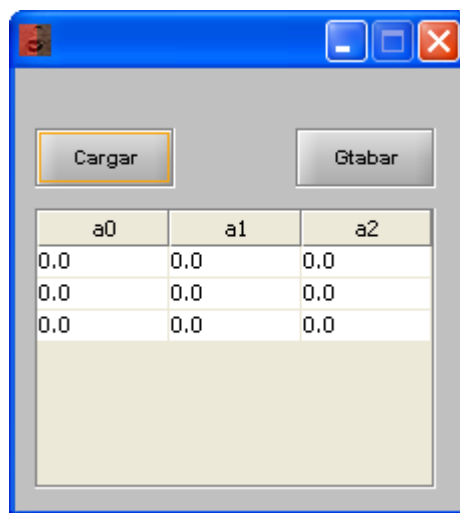


Figura 18

1.10. Escribiendo datos en una matriz

En esta aplicación se trata de generar dos matrices o tablas de tipo 1D y después representarlas gráficamente.

Para ello utilizamos como ejemplo el cálculo de los valores de la función seno de cien valores de ángulo, que son generados mediante el bloque de función “Calculadora” al que le llegan los valores x del Angulo que son generados por el “Reloj”.

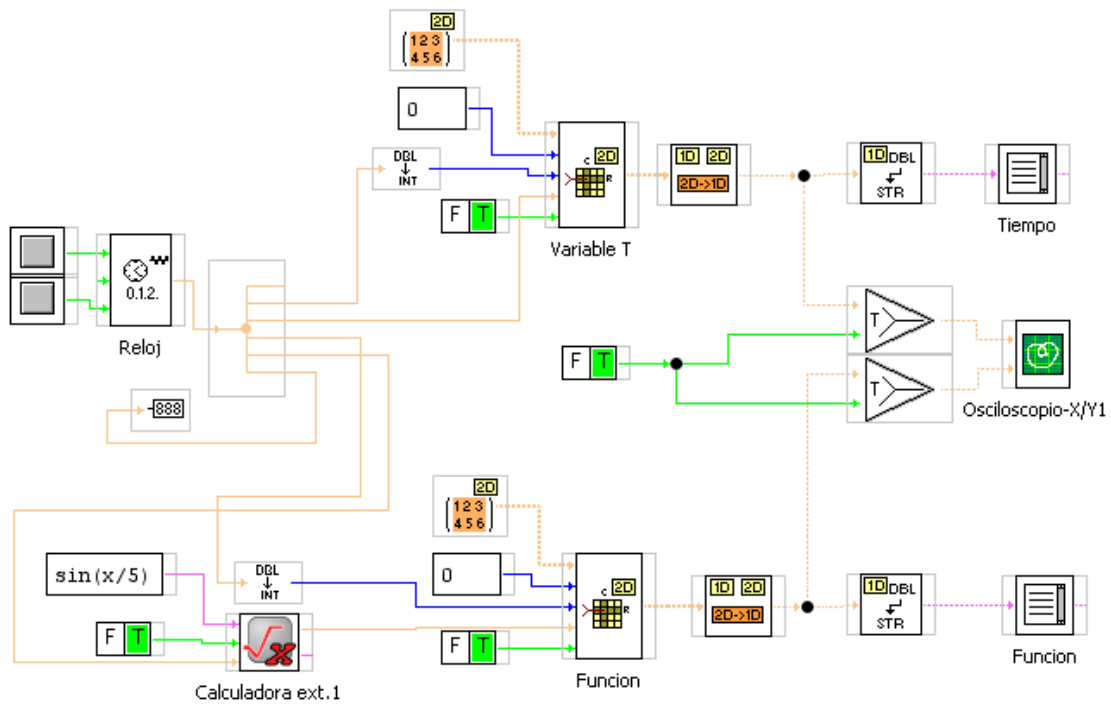
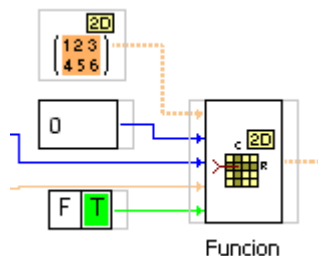


Figura 19

Los valores, “ x ” y “ $\sin(x)$ ” son llevados a sendos bloques de matriz 2D en donde se escriben en secuencia en la columna “0”.



Este bloque entrega los datos a un convertidor de matriz 2D a 1D que después se pasa por un bloque de conversión de tipo de datos para mostrarlo en la Lista correspondiente y por otro (en formato DBL) se lleva al bloque del trazador grafico “Osciloscopio”

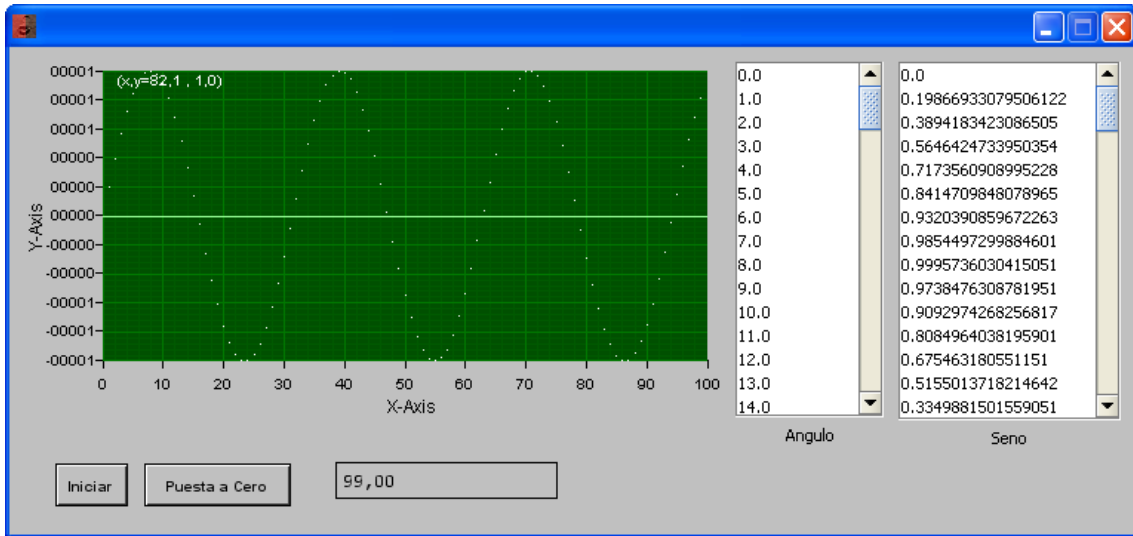


Figura 20

En la figura 20 vemos el resultado de la simulación. Las tablas están cargadas con sus datos y el osciloscopio representa todos cien puntos.

1.11. Matrices 1D (Vectores)

En el siguiente ejemplo (figura 22) se han representado vectores de 1D de distintos tipos de datos.

Se observa que los datos tipo DBL y INT deben ser convertidos a datos STR para poder mostrarlos en las tablas de salida.

Tengamos en cuenta que estos vectores se pueden dimensionar simplemente seleccionando la propiedad "Values" de la ventana de propiedades del Elemento Figura 21

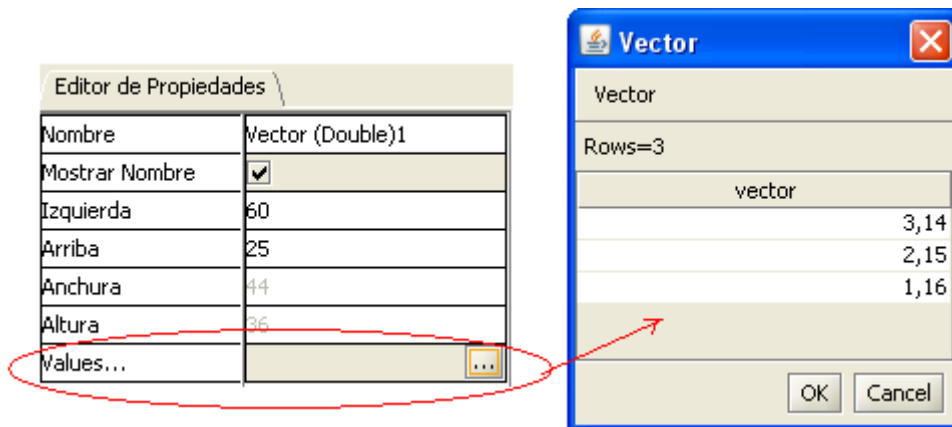


Figura 21

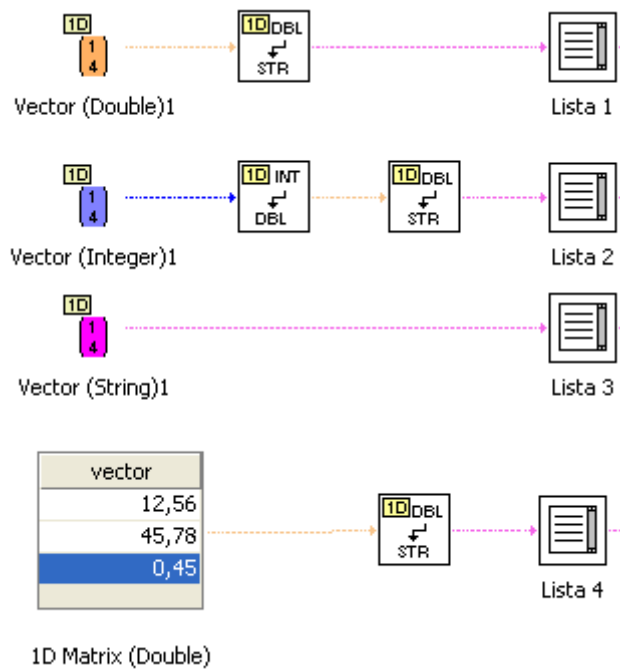


Figura 22

En la figura 23 vemos los datos de salida colocados en cada una de las listas.

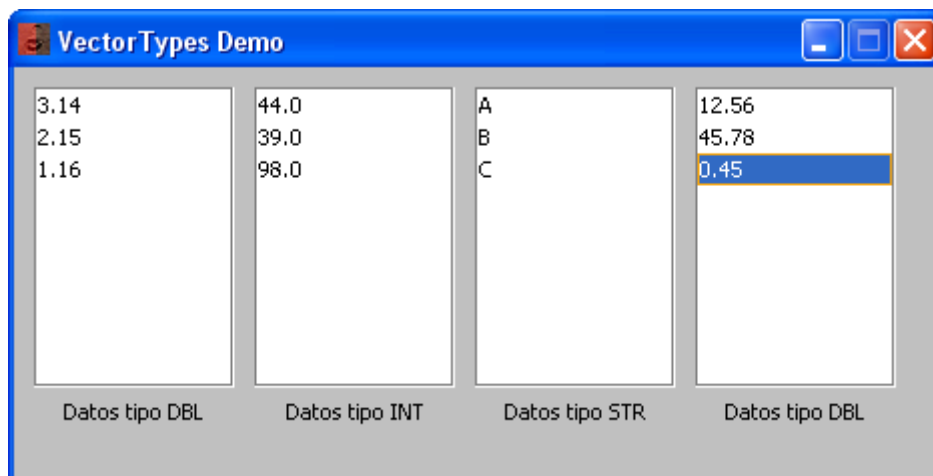


Figura 23