Práctica 6

María del Rosario Aguilar Cruz

Juan Antonio Solís Carrera

LabView es un software que tiene un enfoque de programación gráfica, permite realizar configuración con hardware, proceso de datos y diseño de interfaces de usuario personalizadas, se basa en el diseño de diagrama de bloques permitiendo una mejor visualización cada uno de los aspectos del programa.

Se tiene el panel frontal (interfaz de usuario) y el diagrama de bloques, para esta práctica se desarrollaron 5 programas:

1. Calculadora: se reciben dos números sea enteros o flotantes y se realizan 8 operaciones diferentes suma, resta, multiplicación y división entre los números A y B, se obtiene eleva al cuadrado y se calcula la raíz cuadrada de cada número.
2. Convertidor de temperatura: se recibe la temperatura como un número entero o flotante, se realiza la conversión de °C a °F y viceversa.
3. Generador de número aleatorios: Se generan 10 números aleatorios, se muestra estos números en el panel frontal como un arreglo, y enciende un led indicando cual de ellos es mayor a 0.5.
4. Se expandió este generador de números aleatorios, se obtuvo el tamaño del arreglo, valor mínimo y máximo, posición de estos valores en el arreglo, ordenar arreglo de menor a mayor y de mayor a menor, por último, se da la opción de agregar un número y acomodarlo en la posición deseada.
5. Cluster:

El panel frontal del ejercicio 1 se muestra en la figura 1



Figura 1.- Panel frontal, ejercicio 1

Se tienen 2 controles (A y B) que ayudan a ingresar datos desde la interfaz, y 8 indicadores que muestran los resultados de las operaciones realizada, se tiene un botón de STOP que detiene el programa en el momento que el usuario lo desee. En la figura 2 se muestra el diagrama de bloques de programa.

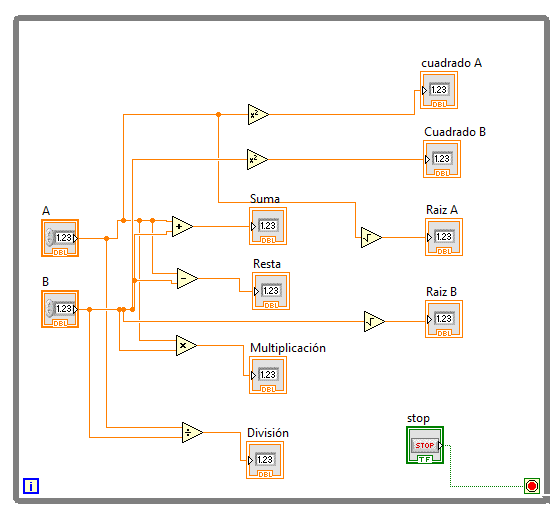


Figura 2.- Diagrama de bloques, ejercicio 1

Se muestra cada una de las conexiones de los dos números ingresados desde el panel frontal a los operadores se pueden encontrar en la paleta de Functions => Mathematics => Numeric, todo el programa se encuentra dentro de un ciclo while que se puede agregar desde Functions => programming => structures, fue utilizado para que el programa se ejecute todo el tiempo que el usuario lo desee, éste se detiene hasta que el botón de STOP (control booleano) sea verdadero.

El panel frontal del ejercicio 2 se muestra en la figura 3.

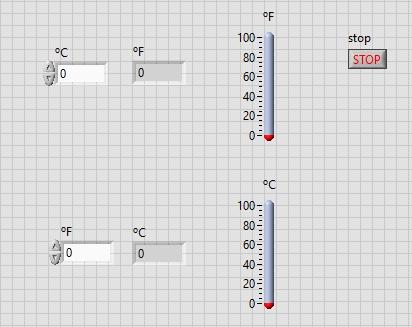


Figura 3.- Panel frontal de ejercicio 2

En el panel frontal se muestran 2 controles numéricos y uno booleano, se tienen 4 indicadores, 2 numéricos y 2 termómetros. En la figura 4 se muestra el diagrama de bloques del ejercicio 2.

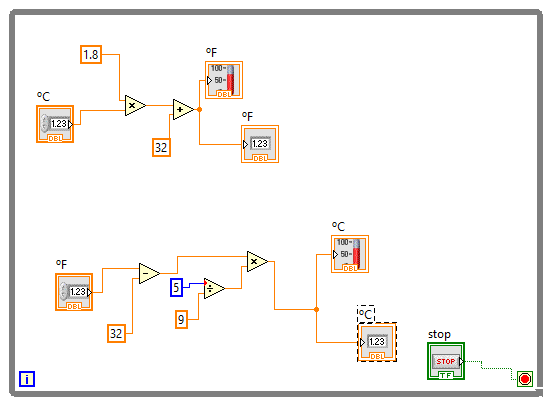


Figura .- Diagrama de bloques de ejercicio 2

Para hacer la conversión de °C a °F, se realizó la operación de la siguiente ecuación

Para el caso de °F a °C se utilizó la siguiente ecuación

El indicador del termómetro se encuentra en la paleta de controles => Num Ins => Thermometer. Todo el programa se encuentra dentro de un while para que sea ejecutado mientras el usuario lo desee.

En la figura 5 se muestra el panel frontal del ejercicio 3.

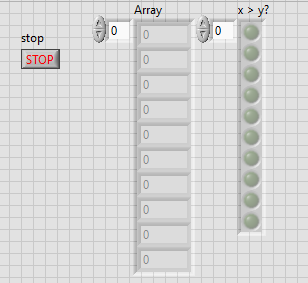


Figura 5.- Panel frontal del ejercicio 3

Se muestran 2 indicadores, uno de leds y el otro es un arreglo, se muestra también un control STOP, que detendrá el programa cuando el usuario lo desee, en la figura 6 se muestra el diagrama de bloques del ejercicio 3.

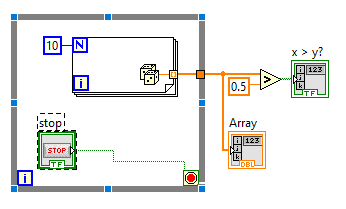


Figura .- Diagrama de bloques del ejercicio 3

Para generar los 10 números aleatorios se utilizó un for, el contador empieza en 1 y llega hasta 10, se utiliza la función de random-number (0-1), para generar números aleatorios, estos son enviados a un indicador de arreglo, por otro lado, se pregunta si el número es mayor a 0.5, si es verdadero se enciende un led, en el panel frontal, para insertar este indicador se crea desde el operador booleano. Debido a que el indicador del arreglo y los led’s se encuentra fuera del while solo se cambiaran los datos cuando el programa se haya detenido.

El panel frontal del ejercicio 4 se muestra en la figura 7.

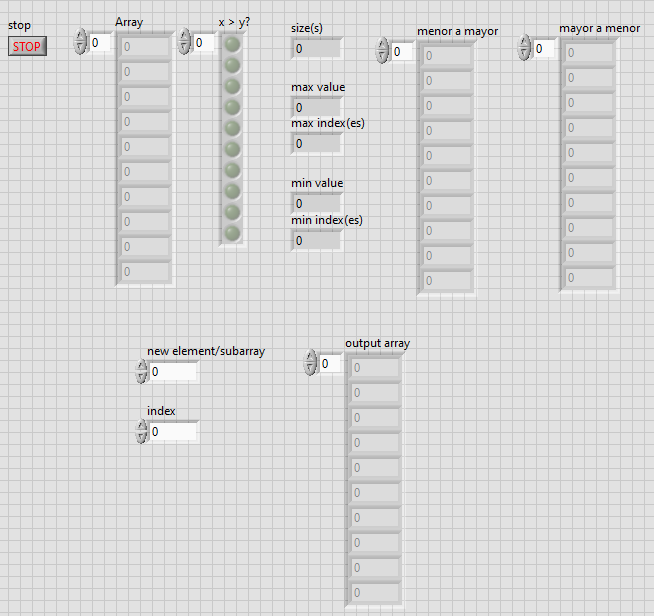


Figura 7.- Panel frontal de ejercicio 4

Se muestran 3 controles, uno es el botón de STOP que detiene el programa, los otros se encargan de agregar un número al arreglo en la posición deseada, 4 de los indicadores son arreglos y 5 muestran datos del primer arreglo, tamaño, valor máximo y la posición en donde se encuentran así también, el valor mínimo. En la figura 8 se muestra el diagrama de bloques del ejercicio 4.

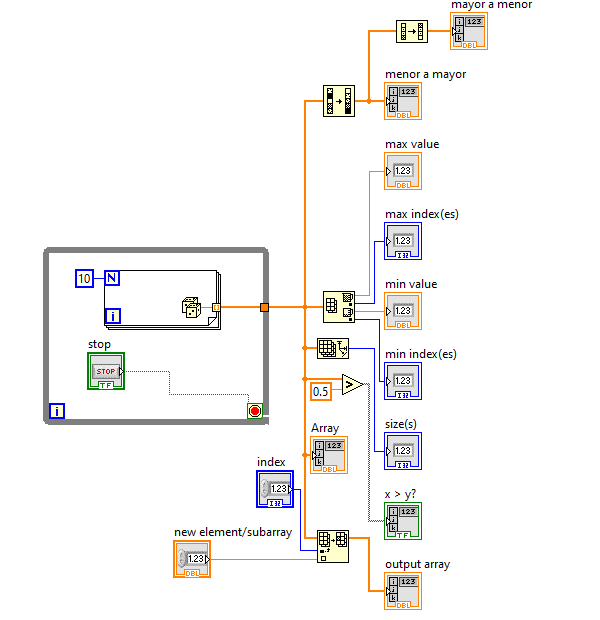


Figura .- Diagrama de bloques de ejercicio 4

Para el primer arreglo, se generaron los números aleatorios como en el ejercicio 3, después para conocer el valor máximo y mínimo se utilizó una función llamada Max & min, que se encuentra en la paleta de Function => programming => array, esta función proporciona también el índice en el cual se encuentra estos valores. Para ordenar el arreglo de menor a mayor se utilizó la función Sort 1D Array y para acomodarlos de mayor a menor se utilizó la función Reverse 1D Array, para agregar un dato en un índice específico se utilizó la función Insert Into Array, para el tamaño del arreglo se utilizó la función Array Size.