

C# program using APLNext Lightweight Array Engine

The APLNext 'Lightweight Array Engine' provides array based operations, analogous to many APL operations. The APLNext LAE is contained in small fully-managed .Net 2.0 assemblies.

In this very simple example a C# console project is used to gather input to create a 'Cielo Variable' array and multiply it by a constant. The SynOps class includes many of the array-based operators of APL extended from scalar to array arguments.

```
using System;
using System.Collections.Generic;
using System.Text;
using VisualCielo.Numeri;
using VisualCielo.Numeri.DataObjects;

namespace LAE_Example_1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("APLNext: LAE: Example 1");

            Console.WriteLine("#Rows [e.g. 2]: ");
            cvar shape = new cvar(0, 0);
            shape = SynOps.AssignKey(shape, new cvar(Convert.ToDouble(Console.ReadLine())), SynOps.sys, new cvar(0));

            Console.WriteLine("#Columns [e.g. 3]: ");
            shape = SynOps.AssignKey(shape, new cvar(Convert.ToDouble(Console.ReadLine())), SynOps.sys, new cvar(1));

            Console.WriteLine("Elements [e.g. 1 2.2 -3.78 4 -5 234]:");
            string[] str_vals = Console.ReadLine().Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);
            double[] dbl_vals = new double[str_vals.Length];
            for (int i = 0; i < dbl_vals.Length; i++)
            {
                dbl_vals[i] = Convert.ToDouble(str_vals[i]);
            }
            cvar vals = new cvar(dbl_vals);
            vals = SynOps.rho(shape, new cvar(dbl_vals), vals);

            Console.WriteLine("Multiplier [e.g. 100]: ");
            cvar mult = new cvar(Convert.ToDouble(Console.ReadLine()));
            vals = SynOps.Times(mult, vals, vals);

            int[] IJ = shape.ConvertToIntVector(shape);
        }
    }
}
```

Create a Cielo Variable (cvar), shape, as a simple 1-dimensional array with elements 0, 0.

Capture the user-entry for the first dimension of the array to be created from the console and store it in the 1st element of the shape cvar object

Capture the user-entry for the second dimension of the array to be created from the console and store it in the 2nd element of the shape cvar object.

Create the Cielo Variable (cvar), vals, to contain the user-entered array element values and shape using the cvar shape.

Capture the user-entered multiplier from the console and store it in the Cielo Variable (cvar), mult.

Use the SynOps.Times operator to perform the multiplication in an array-based manner, i.e. without programming a loop.

```

for (int I = 0; I < IJ[0]; I++)
{
    for (int J = 0; J < IJ[1]; J++)
    {
        cvar val = new cvar();
        val = SynOps.Index(vals, new cvar(I, J), val);
        Console.WriteLine("Item[{0};{1}]: {2}", I, J, val);
    }
}
Console.ReadLine();
}
}
}

```

Use the SynOps.Index operation to prepare the values of the multiplied array for display in the console.

For this specific example, a 2 by 3 numeric array of doubles is created from the user input to the console. The array is then multiplied by 10 and the values of the resulting array are written to the console.

```

file:///C:/Users/Joe.BLAZESSIBIZ/Documents/Visu...
APLNext: LAE: Example 1
#Rows [e.g. 2]:
2
#Columns [e.g. 3]:
3
Elements [e.g. 1 2.2 -3.78 4 -5 234]:
1 2.345 678.90 11.12 13 14.1516
Multiplier [e.g. 100]:
10
Item[0;0]: 10
Item[0;1]: 23.45
Item[0;2]: 678.9
Item[1;0]: 111.2
Item[1;1]: 130
Item[1;2]: 141.516
-

```