LABORATORIO 4 - UTILIZACIÓN DE ARREGLOS

Ejercicio 1

Dado el arreglo {10,20,5,15,30,20}

Informar el arreglo de la forma: "Indice: X, Valor: Y"

Totalizar el arreglo e informar el total

Informar el contenido de las posiciones impares (por ejemplo, las posiciones 1,3,5,etc)

Informar el mayor número

Informar cuántas veces aparece el número 20

Pseudocódigo

```
DEFINIR  vec[10,20,5,15,30,20];
DEFINIR  n,total=0,maximo=vec[0],cont=0
PARA i DESDE 1 HASTA 6 CON i=i+1 HACER
        total <- total + vec[n]
        SI n ES PAR  ENTONCES
                INFORMAR n
                INFORMAR vec[n]
        FIN SI

                SI  vec[n] > maximo ENTONCES
                        maximo <-vec[n]
        FIN SI

        SI vec[n] == 20) ENTONCES
                cont <-cont +1
                FIN SI
        FIN PARA

        INFORMAR total
        INFORMAR maximo
        INFORMAR cont
```

Codificación en C#

```
int[] vec={10,20,5,15,30,20};
int n, total=0, maximo=vec[0], cont=0;
for (n = 0; n < vec.length; n++){
        if(n%2==1)Console.WriteLine("PosImp " + n + " " + " valor:" + vec[n]);
        total = total + vec[n];
        if (vec[n] > maximo) maximo = vec[n];
        if (vec[n] == 20) cont++;
}
```

```
Console.WriteLine("El mayor es: " + maximo);
Console.WriteLine("El total es: " + total);
Console.WriteLine("El 20 apareció: " + cont+" veces");
```

Prueba de escritorio

ciclo	*n*	*vec[n]*	*total*	*mayor*	*cont*
1	*1*	*10*	*10*	*10*	*0*
2	*2*	*20*	*30*	*20*	*1*
3	*3*	*5*	*35*	*20*	*1*
4	*4*	*15*	*50*	*20*	*1*
5	*5*	*30*	*80*	*30*	*1*
6	*6*	*20*	*100*	*30*	*2*

Ejercicio 2

Dado el arreglo inflación {0.8, 0.1, 0.3, 0.4, 0.3, 0.6, 0.5, 0.3, 0.7, 0.3, 0.2, 0.9}

Cada ítem del arreglo representa la inflación de un mes de tal manera que el primer ítem del arreglo que es 0.8 representa la inflación de Enero, y el último ítem del arreglo que es 0.9 representa la inflación de diciembre.

Se pide:

Informar la inflación anual

Informar la inflación más baja, junto con el número de mes. Por ejemplo: Mes 2 = 0.1

Informar la inflación más alta, junto con el número de mes. Por ejemplo: Mes 12 = 0.9

Informar el promedio de inflación (inflación total / 2)

Pseudocódigo

```
DEFINIR  inflacion={0.8,0.1,0.3,0.4,0.3,0.6,0.5,0.3,0.7,0.3,0.2,0.9}
DEFINIR   total,maximo=inflacion[0],minimo=inflacion[0];
DEFINIR  n,int mes_min=0,mes_max=0
PARA n DESDE 1 HASTA 12 CON n=n+1 HACER

total <- total + inflacion[n]
SI  inflacion[n] > maximo ENTONCES
                        maximo <-inflacion[i]
mes_max<-n+1
FIN SI
SI  inflacion[n] < minimo ENTONCES
                        minimo <-inflacion[n]
```

mes_min<ni+1

FIN SI

FIN PARA

INFORMAR total

INFORMAR promedio

INFORMAR maximo y mes_max

INFORMAR maximo y mes_min

Codificación en C#

```
double[] inflacion={0.8,0.1,0.3,0.4,0.3,0.6,0.5,0.3,0.7,0.3,0.2,0.9};
 int mes_min=0,mes_max=0;
int n,double total=0,maximo=inflacion[0], minimo=inflacion[0];
for (n = 0; n < infla.length; n++){
        total = total + inflacion[n];//total+=vec[n]
        if (inflacion[n] > maximo){
                maximo = inflacion[n];
                mes_max=n+1;
                }
                if (inflacion[n] < minimo){
                minimo = inflacion[n];
                mes_min=n+1;
        }
}
Console.WriteLine("Promedio: " + total/12);
Console.WriteLine("Mínima inflación: " + minimo+" mes:"+mes_min);
Console.WriteLine("Máxima inflación: " + maximo+" mes:"+mes_max);
```

Prueba de escritorio

n	***infla[n]***	***mes***	***total***	***maximo***	***mínimo***
0	***0.8***	***1***	***0.8***	***0.8***	***0.8***
1	***0.1***	***2***	***0.9***	***0.8***	***0.1***
2	***0.3***	***3***	***1.2***	***0.8***	***0.1***
3	***0.4***	***4***	***1.6***	***0.8***	***0.1***
4	***0.3***	***5***	***1.9***	***0.8***	***0.1***
5	***0.6***	***6***	***2.5***	***0.8***	***0.1***
6	***0.5***	***7***	***3.0***	***0.8***	***0.1***
7	***0.3***	***8***	***3.3***	***0.8***	***0.1***

8	***0.7***	***9***	***4.0***	***0.8***	***0.1***
9	***0.3***	***10***	***4.3***	***0.8***	***0.1***
10	***0.2***	***11***	***4.5***	***0.8***	***0.1***
11	***0.9***	***12***	***5.4***	***0.9***	***0.1***

Ejercicio 3

Agregar los cambios que resulten necesarios al ejercicio anterior para que en los puntos b y c se informe la inflación junto con el nombre del mes. Por ejemplo: Mes FEBRERO = 0.1

Codificación en C#

```
double[] inflacion={0.8,0.1,0.3,0.4,0.3,0.6,0.5,0.3,0.7,0.3,0.2,0.9};

String[]mes={"enero","febrero","marzo","abril","mayo","junio","julio","agosto","septiembre","octubre","noviembre","diciembre"};
String mes_min="",mes_max="";
int n,double total=0,maximo=inflacion[0], minimo=inflacion[0];
for (n = 0; n < inflacion.length; n++){
        total = total + inflacion[n];//total+=vec[n]
        if (infla[n] > maximo){
                maximo = inflacion[n];
                mes_max=mes[n];
        }
        if (inflacion[n] < minimo){
                minimo = inflacion[n];
                mes_min=mes[n];
        }
}
Console.WriteLine("Promedio: " + total/12);
Console.WriteLine("Mínima inflación: " + minimo+" mes:"+mes_min);
Console.WriteLine("Máxima inflación: " + maximo+" mes:"+mes_max);
```

Ejercicio 4

Uso de arreglo con ingreso de datos por teclado

Ingrese por teclado la facturación de los últimos 6 meses. Solo se pueden ingresar números.

Informar:

a) la facturación total

b) el promedio de facturación

c) la máxima facturación

d) la máxima facturación

Pseudocódigo

```
DEFINIR  facturacion[]
```

```
DEFINIR  n, total,promedio,maximo,minimo;

PARA n DESDE 1 HASTA 6 CON n=n+1 HACER

leer facturacion[n]

total <- total + facturacion[n]
FIN PARA

PARA n DESDE 1 HASTA 6 CON n=n+1 HACER
SI facturacion[n] > maximo ENTONCES
                    maximo <-facturacion[n]

FIN SI
SI  facturacion[n] < minimo ENTONCES
                    minimo <-inflacion[n]

FIN SI


FIN PARA

INFORMAR total
INFORMAR promedio
INFORMAR maximo
INFORMAR maximo
```

Codificación en C#

```
double[] facturacion=new double [6];
int n,double total=0,maximo=0, minimo=infla[0];

// Ingreso de las 6 facturaciones
for (n = 0; n < facturacion.length; n++){
        Console.WriteLine("Ingrese facturación: ");
        facturacion[n] = Double.parseDouble(teclado.next());
}

// Proceso de los datos

for (n = 0; n < facturacion.length; n++){
        total = total + facturacion[n];//total+=facturacion[n]
        if (facturacion[n] > maximo){
                maximo = facturacion[n];
        }

        if (facturacion[n] < minimo){
                minimo = facturacion[n];
        }
```

```
}
Console.WriteLine("Facturación total:  " + total);
Console.WriteLine("Promedio: " + facturacion.length/12);
Console.WriteLine("Mínima inflación: " + minimo);
Console.WriteLine("Máxima inflación: " + maximo);
```

Prueba de escritorio

n	*facturacion[n]*	*mes*	*total*	*maximo*	*mínimo*
0	*200*	*1*	*200*	*200*	*200*
1	*300*	*2*	*500*	*300*	*200*
2	*250*	*3*	*750*	*300*	*200*
3	*350*	*4*	*1100*	*350*	*200*
4	*200*	*5*	*1300*	*350*	*200*
5	*180*	*6*	*1480*	*350*	*180*

Ejercicio 5

Copiar el contenido del arreglo origen al arreglo destino utilizando estructura de control de flujo repetitiva, y luego informar el índice y los valores del nuevo arreglo

Pseudocódigo

```
DEFINIR  origen[ ]={2,10,-4,8,0}
DEFINIR  destino[ ]
DEFINIR  n

PARA n DESDE 1 HASTA 5 CON n=n+1 HACER

origen[n]<-destino[n]

FIN PARA

PARA n DESDE 1 HASTA 5 CON n=n+1 HACER

INFORMAR n
INFORMAR   destino[n]

FIN PARA
```

Codificación en C#

```
int origen[ ] = {2,10,-4,8,0};
int [ ] destino = new int [origen.length];
for (int n = 0; n < origen.length; n++){
        destino[n]=origen[n];
}
for (int n = 0; n < origen.length; n++){
        Console.WriteLine("Índice: " + n + " valor: " + destino[n]);
}
```

Prueba de escritorio

número de ciclo	***n***	***origen[n]***	***destino[n]***
1	***1***	***2***	***2***
2	***2***	***10***	***10***
3	***3***	***-4***	***-4***
4	***4***	***8***	***8***
5	***5***	***0***	***0***

BONUS

Ejercicio 6

Copiar el contenido del arreglo origen al arreglo destino, dejando en este último los valores invertidos respecto del arreglo origen. Utilizar estructura de control de flujo repetitiva, y luego informar el índice y los valores del nuevo arreglo.

Por ejemplo, informar de la siguiente manera:

Arreglo origen

0 1002

1 104

2 309

3 500

Arreglo destino

0 500

1 309

2 104

3 1002

Pseudocódigo

DEFINIR origen[ ]={2,10,-4,8,0}
DEFINIR destino[ ]
DEFINIR n

PARA n DESDE 1 HASTA 5 CON n=n+1 HACER

destino[n] = origen[origen.Length -n-1]

FIN PARA

PARA n DESDE 1 HASTA 5 CON n=n+1 HACER

INFORMAR n
INFORMAR destino[n]

FIN PARA

Codificación en C#

```
int [ ]origen = {2,10,-4,8,0};
int [ ] destino = new int [origen.length];
for (int n = 0; n < origen.length; n++){
        destino[n] = origen[origen.length - n-1];
}
for (int n = 0; n < origen.length; n++){
        Console.WriteLine("Índice: " + n + " valor: " + destino[n]);
}
```

Prueba de escritorio

número de ciclo	***n***	***origen[n]***	***destino[n]***
1	***1***	***2***	***0***
2	***2***	***10***	***8***
3	***3***	***-4***	***-4***
4	***4***	***8***	***10***
5	***5***	***0***	***2***

Ejercicio 7

Dado un arreglo del 0 al 11 con la facturación correspondiente a todo un año, informar la facturación por trimestre. Para esto,armar un

arreglo de 4 posiciones, donde cada posición contenga la facturación de cada trimestre.

Pseudocódigo

```
DEFINIR  facturacion[ ]
DEFINIR  trimestre[ ]
DEFINIR  n, total1,total2,total3,total4

PARA n  DESDE 1 HASTA 12 CON n=n+1 HACER


total <- total + facturacion[n]
SI facturacion[n] <=2 ENTONCES
                        total1<-total1+ facturacion[n]

FIN SI
SI facturacion[n]>=3 Y facturacion[n]<=5  ENTONCES
                        total2<-total2+ facturacion[n]

FIN SI
SI facturacion[n]>=6 Y facturacion[n]<=8 ENTONCES
                        total3<-total3+ facturacion[n]

FIN SI
SI facturacion[n]>=9 ENTONCES
                        total4<-total4+ facturacion[n]

FIN SI


FIN PARA

trimestre[0] <- total1
trimestre[1] <- total2
trimestre[2] <- total3
trimestre[3] <- total4
PARA n DESDE 1 HASTA 4 CON n=n+1 HACER
INFORMAR trimestre
INFORMAR facturacion
FIN PARA
```

Codificación en C#

```
int[ ] facturacion = { 80, 20, 50, 40, 60, 70, 80, 30, 50, 20, 40, 60 };
int[ ] trimestre = new int[4];
int total1 = 0, total2 = 0, total3 = 0, total=4;
```

```
for (int n = 0; n < facturacion.length; n++){
	if (n <= 2){
		total1 = total1 + facturacion[n];

	}
	if (n >= 3 & n <= 5){
		total2 = total2 + facturacion[n];
	}
	if (n >= 6 & n <= 8){
		total3 = total3 + facturacion[n];
	}
	if (n >= 9){
		total4 = total4 + facturacion[n];
	}
}

trimestre[0] = total1;
trimestre[1] = total2;
trimestre[2] = total3;
trimestre[3] = total4;

for (int n = 0; n < trimestre.length; n++){
	Console.WriteLine("Trimestre: " + n + " Facturacion: " + trimestre[n]);
}
```

Prueba de escritorio

Número de ciclo	**n**	**facturacion[n]**	**total1**	**total2**	**total3**	**total4**	**trimeste[n]**
1	1	80	80	0	0	0	
2	2	20	100	0	0	0	
3	3	50	150	0	0	0	330
4	4	40	150	40	0	0	
5	5	60	150	100	0	0	
6	6	70	150	170	0	0	170
7	7	80	150	170	80	0	
8	8	30	150	170	110	0	
9	9	50	150	170	160	0	160
10	10	20	150	170	160	20	

11	11	40	150	170	160	60	
12	12	60	150	170	160	120	120