



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA INDUSTRIAL

PRÁCTICA Nro. 1 DE ALGORITMOS Y ESTRUCTURAS DE DATOS 2014-2
15 Setiembre 2014

TIEMPO: 1 Hr. 50Min.

Indicaciones

- 1) Resuelva cada pregunta en páginas diferentes.
- 2) Cualquier función adicional que requiera en su solución; debe escribirla necesariamente.
- 3) No se aceptarán reclamos en pruebas desarrollados con lápiz ni con correctores líquidos.
- 4) Queda prohibido el uso de celulares y/o cualquier equipo multimedio de comunicación.

PREGUNTA 1:

Se pide implementar las funciones sobrecarga llamada *Gestionar* de acuerdo a lo siguiente:

- a) (3 puntos) *void Gestionar(int d, int m)*. - En el parámetro *d* y *m* recibe el día y el mes (ya validado) correspondiente al año 2014, con esta información debe mostrar el nombre de la estación del año a la que pertenece esta fecha. Se brinda la siguiente información adicional.

ESTACION	FECHA INICIO- TERMINO(dia/mes)
Primavera	21/09 al 20/12
Verano	21/12 al 20/03
Otoño	21/03 al 20/06
Invierno	21/06 al 20/09

- b) (4 puntos) *void Gestionar(int x[][N], int a, int n)*. - *x* es una matriz cuadrada de orden *n*, en *x* se almacena números primos, para ello el numero entero de partida es *a*, donde este debe ser convertido a su equivalente binario y almacenar como primer elemento de la matriz, el siguiente número a procesar será *a+1* y se almacena en binario como siguiente elemento de la matriz y así sucesivamente. Un numero binario se obtiene por divisiones sucesivas entre 2 y se toma para formar el binario los residuos pero en forma inversa. Como ejemplo, sea *a = 23*, entonces en binario es 10111

PREGUNTA 2:

Examine el siguiente programa y su salida o ejecución, Implemente solo aquellas funciones donde se indica puntaje.

```
Hexadecimal: 2d7b
Decimal: 11643
Decimales en el vector: 9146 29210 76 15 13534 11643
Process returned 0 (0x0)   execution time : 0.141 s
Press any key to continue.
```

```
#include <iostream>
#include <string.h>
#define N 255
#define M 50
int Dig(char c); /*(2 puntos) Devuelve el decimal equivalente al caracter hexadecimal recibido como parametro. Por ejemplo: Dig('d') devuelve 13 */
int Pasar_A_Decimal(char Hex[]); /*(2 puntos) Devuelve el decimal que representa la cadena Hex */
void PrintVector(int v[], int n); //Imprime un vector de enteros
void Cargar_Numeros(char nums_hex[N], int Deci[M], int &n); /* (3puntos) Recibe la cadena de números hexadecimales y Carga en el arreglo Deci los decimales equivalentes, n es el tamaño del arreglo */
using namespace std;
int main()
{
    char hex[] = "2d7b";
    cout << "Hexadecimal: " << hex << endl;
    cout << "Decimal: " << Pasar_A_Decimal( hex ) << endl;
    char Hexas[N] = "23ba 721a 4c f 34de 2d7b";
    int Deci[M], n;
    Cargar_Numeros( Hexas, Deci, n );
    PrintVector( Deci, n );
    return 0;
}
```


PREGUNTA 3: (6 puntos)

Dado el siguiente programa y una posible ejecución:

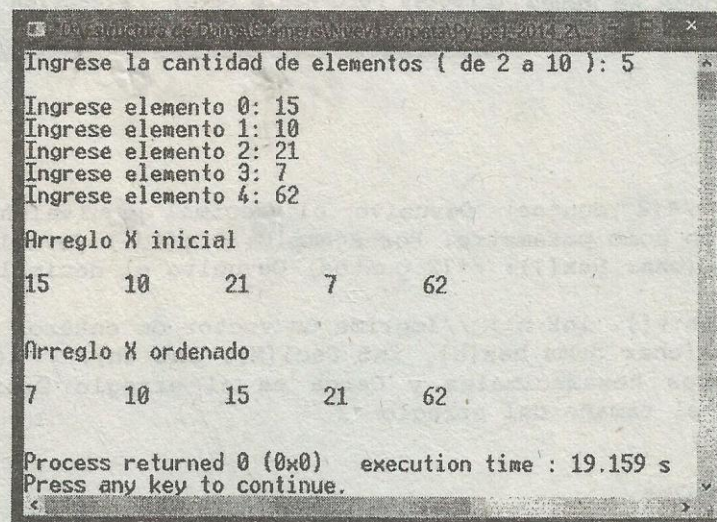
```
#include <iostream>
#include "pcl_2014_2.h"

using namespace std;

int main( )
{
    ARREGLO_ENTEROS(x,10);
    int ne,i,j;
    do
    {
        cout << "Ingrese la cantidad de elementos ( de 2 a 10 ): ";
        LEER_ENTERO(ne);
    } while ( NO_VALIDO(ne,2,10) );
    cout << endl;
    for( i = 0 ; i < ne ; i++ )
    {
        cout << "Ingrese elemento " << i << ": ";
        LEER_ENTERO(x[i]);
    }

    cout << endl << "Arreglo X inicial" << endl << endl;
    MOSTRAR_ARREGLO( x , ne );
    for( i = 0 ; i < ne-1 ; i++ )
        for( j = i+1 ; j < ne ; j++ )
            SI_MAYOR(x,i,j)
                INTERCAMBIAR(x,i,j);
    cout << endl << "Arreglo X ordenado" << endl << endl;
    MOSTRAR_ARREGLO( x , ne );
    return 0 ;
}
```

Escriba el contenido del archivo de inclusión `pcl_2014_2.h` para el correcto funcionamiento del programa.



```
cmd.exe - C:\Program Files\Microsoft Visual Studio 9.0\VC\bin\amd64\cl.exe
Ingrese la cantidad de elementos ( de 2 a 10 ): 5
Ingrese elemento 0: 15
Ingrese elemento 1: 10
Ingrese elemento 2: 21
Ingrese elemento 3: 7
Ingrese elemento 4: 62

Arreglo X inicial
15    10    21    7    62

Arreglo X ordenado
7     10    15    21    62

Process returned 0 (0x0)   execution time : 19.159 s
Press any key to continue.
```