

Apellido y nombre: _____ Legajo: _____ Cursó con Prof: _____

Cantidad de hojas entregadas: _____ Nota: _____ Evaluó Prof: _____

- Si luego de la lectura del examen, durante la resolución tiene alguna duda, escriba hipótesis de trabajo, las cuales también serán evaluadas.
- Los puntos que solicitan codificación puede ser respondidos en C, C++, o en Pascal, pero debe indicar el lenguaje utilizado.
- En C y C++ prototipo refiere a la declaración de la función, es decir tipo de dato retornado, nombre de la función, y tipos de los parámetros; en Pascal el concepto prototipo es análogo al encabezado del procedimiento o de la función previa al bloque que lo define.

Distancias entre localidades

Temas evaluados: Resolución de problemas, estructuras de datos, matrices, y lenguaje de programación

Contexto

Usted es miembro de un equipo de desarrollo de software. El equipo debe desarrollar un sistema para **consultar distancias y rutas entre localidades**. Cada localidad se representa con un número entre 1 y 100, y las distancias y rutas con enteros positivos. Usted es responsable de realizar las **consultas** sobre los datos que ya están en memoria.

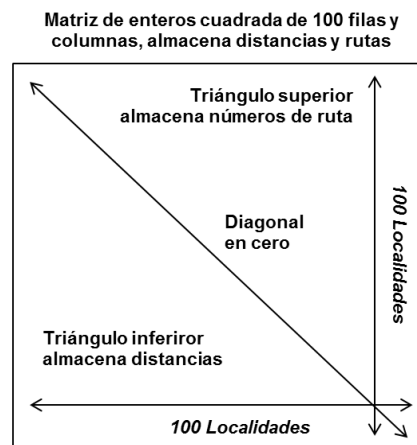
Problema

Desarrollar las funciones que permiten consultar información acerca de rutas entre localidades; utilizar los recursos disponibles.

Recursos disponibles

- Vector (arreglo de una dimensión) **Rutas**. Contiene la distancia en kilómetros entre todas las localidades que están relacionadas por una ruta directa. Cada ruta se representa como una 4-upla: **localidad₁** (1..100), **localidad₂** (1..100), **distancia** (entero positivo) y **ruta** (entero positivo).
Una ruta está descripta solo una vez; por ejemplo, si la localidad α está a 10 km de la localidad β y los une la ruta 5, entonces, el arreglo contiene la 4-upla $(\alpha, \beta, 10, 5)$, ó la 4-upla $(\beta, \alpha, 10, 5)$, pero no ambas.
- Matriz (arreglo de dos dimensiones) **M**. Es una matriz de enteros cuadrada de tamaño 100, contiene las *distancias* y números de *rutas* entre las localidades. Por ejemplo, para las localidades i y j , M_{ij} representa la distancia de la localidad i a la j , mientras que M_{ji} representa el número de ruta que conecta j con i , y M_{ii} y M_{jj} están en cero. Recuerde que:
 1. Si M_{ij} pertenece al triángulo inferior, entonces $i > j$ (distancia)
 2. El elemento simétrico respecto la diagonal principal es M_{ji} (ruta)

[0][0]	[0][1]	[0][2]	[0][3]
[1][0]	[1][1]	[1][2]	[1][3]
[2][0]	[2][1]	[2][2]	[2][3]
[3][0]	[3][1]	[2][2]	[3][3]



Se pide

1. **Codifique** todas las declaraciones para el arreglo **Rutas**.
2. **Codifique** el prototipo (no la definición) de la función **InicializarM** que inicializa **M** con los datos necesarios para resolver las consultas. Decida los **parámetros** y **justifique** su decisión.
3. **Diseño o codifique** la función **MostrarRuta** que dado **tres** parámetros **localidad₁**, **localidad₂**, y **M**, muestra por *cout* (la salida estándar) el identificador de la ruta y la distancia que une **localidad₁** con **localidad₂**. Si no hay ruta, muestra un mensaje apropiado.
4. **Diseño o codifique** la función **MostrarVecinas**. Dada **dos** parámetros **localidad**, y **M**, muestra por *cout* las localidades con las que **localidad** tiene ruta directa. Si no hay vecinas, muestra un mensaje apropiado.

Apellido y nombre: _____ Legajo: _____ Curso con Prof: _____

Cantidad de hojas entregadas: _____ Nota: _____ Evaluó Prof: _____

- Si luego de la lectura del examen, durante la resolución tiene alguna duda, escriba hipótesis de trabajo, las cuales también serán evaluadas.
- Los puntos que solicitan codificación puede ser respondidos en C, C++, o en Pascal, pero debe indicar el lenguaje utilizado.
- En C y C++ prototipo refiere a la declaración de la función, es decir tipo de dato retornado, nombre de la función, y tipos de los parámetros; en Pascal el concepto prototipo es análogo al encabezado del procedimiento o de la función previa al bloque que lo define.

Distancias entre localidades

Temas evaluados: Resolución de problemas, estructuras de datos, matrices, y lenguaje de programación

Contexto

Usted es miembro de un equipo de desarrollo de software. El equipo debe desarrollar un sistema para **consultar distancias y rutas entre localidades**. Cada localidad se representa con un nombre (cadena de 20 caracteres), y las distancias y rutas con enteros positivos. Usted es responsable de tomar los datos desde un flujo binario y llevar los a memoria en una representación que facilite las consultas.

Problema

Desarrollar las funciones que permiten llevar a memoria los datos disponibles en archivos binarios para permitir, luego, consultar información acerca de rutas entre localidades; utilizar los recursos disponibles.

Recursos disponibles

- Un flujo de registros (archivo abierto) **Rutas**. Contiene la distancia en kilómetros entre todas las localidades que están relacionadas por una ruta directa. Cada ruta se representa como una 4-upla: **localidad₁** (Cadena 20 caracteres), **localidad₂** (Cadena de 20 caracteres), **distancia** (entero positivo) y **ruta** (entero positivo).
Una ruta está descripta solo una vez; por ejemplo, si la localidad α está a 10 km de la localidad β y los une la ruta 5, entonces, el arreglo contiene la 4-upla $(\alpha, \beta, 10, 5)$, ó la 4-upla $(\beta, \alpha, 10, 5)$, pero no ambas.
- Una matriz (arreglo de dos dimensiones) **M**. Es una matriz de enteros cuadrada de tamaño 100, no está inicializada.
- Un vector (arreglo) **Localidades** con los nombres de cada una de las 100 ciudades, ordenados alfabéticamente en forma creciente. La posición de la localidad en este vector, determina la fila y columna en **M**. Por ejemplo, si la localidad “San Pedro” está en la posición 21 del vector, entonces en la fila 21 y en la columna 21 están las distancias y rutas de las vecinas de “San Pedro”.
- Una función **BuscarLocalidad** que dado el nombre de una localidad y el vector localidades, retorna la posición en ese vector. La función está disponible, pero usted debe definir su prototipo.

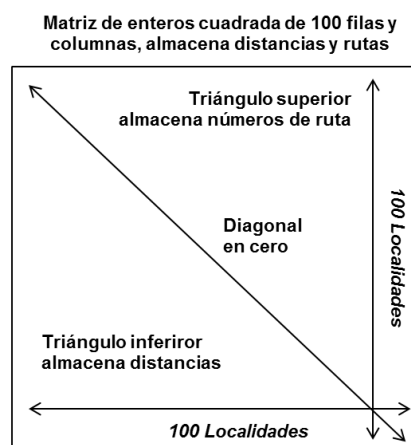
Se pide

1. **Codifique** las todas las **declaraciones** necesarias para **Rutas**, **M**, y **Localidades**.
2. **Codifique** el **prototipo** (no la definición) de la función **BuscarLocalidad**. Decida los **parámetros** y **justifique** su decisión.
3. **Diagrame o codifique** la función **InicializarM** que dados el flujo binario **Ruta**, el vector **Localidades**, y la matriz **M**, cargue en **M** los datos del flujo, y establezca en **cero** los **elementos de M no utilizados**.

La función debe dejar cargar en **M** las **distancias** y números de **rutas** entre las localidades. Por ejemplo, para las localidades **i** y **j**, **M_{ij}** debe representar la distancia de la localidad **i** a la **j**, mientras que **M_{ji}** debe representar el número de ruta que conecta **j** con **i**, y **M_{ii}** y **M_{jj}** debe estar en cero. Como las localidades en el flujo son representadas como cadenas y en la matriz como índices, es necesario en el transformar la representación, para eso debe **invocar** la función **BuscarLocalidad** con el vector **Localidades**. Tenga en cuenta que:

Si **M_{ij}** pertenece al triángulo inferior, entonces $i > j$ (distancia)
El elemento simétrico respecto la diagonal principal es **M_{ji}** (ruta)

[0][0]	[0][1]	[0][2]	[0][3]
[1][0]	[1][1]	[1][2]	[1][3]
[2][0]	[2][1]	[2][2]	[2][3]
[3][0]	[3][1]	[2][2]	[3][3]



Apellido y nombre: _____ Legajo: _____ Cursó con Prof: _____

Cantidad de hojas entregadas: _____ Nota: _____ Evaluó Prof: _____

- Si luego de la lectura del examen, durante la resolución tiene alguna duda, escriba hipótesis de trabajo, las cuales también serán evaluadas.
- Los puntos que solicitan codificación puede ser respondidos en C, C++, o en Pascal, pero debe indicar el lenguaje utilizado.
- En C y C++ prototipo refiere a la declaración de la función, es decir tipo de dato retornado, nombre de la función, y tipos de los parámetros; en Pascal el concepto prototipo es análogo al encabezado del procedimiento o de la función previa al bloque que lo define.

Solucion primera fecha 18/07

```
/* UTN FRBA AED
 * Final 20140718 - Rutas
 * OB&JMS
 */
```

```
#include <stdio.h> // printf
#define LOCS 100 // Cantidad de localidades.
```

```
/* 1. Codifique todas las declaraciones para el arreglo Rutas. */
// typedef tipo Nuevo nombre crea un sinónimo con el nombre Nuevo de un tipo.
typedef struct {
    unsigned localidad1; // int es válido.
    unsigned localidad2;
    unsigned distancia;
    unsigned ruta;
} Ruta;
```

```
Ruta Rutas[LOCS*LOCS];
```

```
/* 2. Codifique el prototipo función InicializarM */
/* La función no retorna valores, por eso es void. El parámetro m tiene la
referencia de la matriz a inicializar, es un parámetro out, ya que va a ser
utilizada para almacenar la información de r de una forma que facilite las
consultas. El parámetro r tiene la referencia a las rutas, como es un parámetro in,
se lo califica con const. Se decide no pasar el tamaño de R porque se deriva de
LOCS.
*/
```

```
void InicializarM(unsigned m[][LOCS], const Ruta r[]);
```

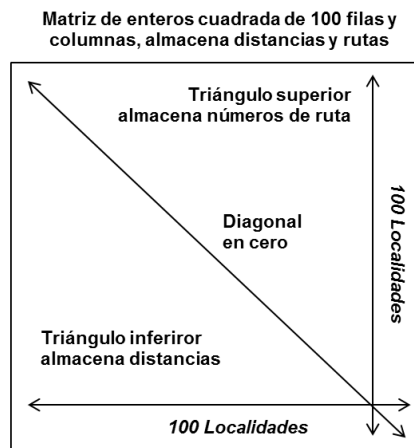
Apellido y nombre: _____ Legajo: _____ Cursó con Prof: _____

Cantidad de hojas entregadas: _____ Nota: _____ Evaluó Prof: _____

- Si luego de la lectura del examen, durante la resolución tiene alguna duda, escriba hipótesis de trabajo, las cuales también serán evaluadas.
- Los puntos que solicitan codificación puede ser respondidos en C, C++, o en Pascal, pero debe indicar el lenguaje utilizado.
- En C y C++ prototipo refiere a la declaración de la función, es decir tipo de dato retornado, nombre de la función, y tipos de los parámetros; en Pascal el concepto prototipo es análogo al encabezado del procedimiento o de la función previa al bloque que lo define.

/*3. Diseñe o codifique la función MostrarRuta

```
void MostrarRuta(unsigned localidad1, unsigned localidad2, unsigned m[][LOCS])
{
```



```
// Determino si está en la diagonal principal
if(localidad1==localidad2){
    printf("Ya está en %d\n", localidad1);
    return;
}

// Determino si hay ruta
int ruta = m[localidad1][localidad2];
if(ruta == 0){
    printf("No hay ruta entre %d y %d\n", localidad1, localidad2);
    return;
}

int distancia = m[localidad2][localidad1]; //en el simétrico a ruta.

// Determino si m[localidad1][localidad2] es triangulo inferior.
if(localidad2<localidad1){
    // En triángulo inferior ruta tiene la distancia, intercambio.
    int temp=ruta;
    ruta=distancia, distancia=temp;
}

// Informo
printf("La localidad %d está a %d km de la localidad %d, y se unen por la
ruta %d\n", localidad1, distancia, localidad2, ruta);
}
```

UTN – FRBA – Algoritmos y Estructura de Datos – Examen Final – 01/08/2014

Apellido y nombre: _____ Legajo: _____ Cursó con Prof: _____

Cantidad de hojas entregadas: _____ Nota: _____ Evaluó Prof: _____

- Si luego de la lectura del examen, durante la resolución tiene alguna duda, escriba hipótesis de trabajo, las cuales también serán evaluadas.
- Los puntos que solicitan codificación puede ser respondidos en C, C++, o en Pascal, pero debe indicar el lenguaje utilizado.
- En C y C++ prototipo refiere a la declaración de la función, es decir tipo de dato retornado, nombre de la función, y tipos de los parámetros; en Pascal el concepto prototipo es análogo al encabezado del procedimiento o de la función previa al bloque que lo define.

/*4. Diseñe o codifique la función MostrarVecinas. Dada dos parámetros localidad, y M, muestra por cout las localidades con las que localidad tiene ruta directa. Si no hay vecinas, muestra un mensaje apropiado.

*/

Las dejamos para ustedes. Gracias profesor Sola...

Apellido y nombre: _____ Legajo: _____ Cursó con Prof: _____

Cantidad de hojas entregadas: _____ Nota: _____ Evaluó Prof: _____

- Si luego de la lectura del examen, durante la resolución tiene alguna duda, escriba hipótesis de trabajo, las cuales también serán evaluadas.
- Los puntos que solicitan codificación puede ser respondidos en C, C++, o en Pascal, pero debe indicar el lenguaje utilizado.
- En C y C++ prototipo refiere a la declaración de la función, es decir tipo de dato retornado, nombre de la función, y tipos de los parámetros; en Pascal el concepto prototipo es análogo al encabezado del procedimiento o de la función previa al bloque que lo define.

Solución segunda fecha

```
/* UTN FRBA AED
 * Final 20140801 - Rutas
 * OB&JMS&HC
 */
```

```
#include <stdio.h> //
#define LOCS 100 // Cantidad de localidades.
```

```
/* 1. Codifique todas las declaraciones. */
```

```
// Registro del flujo
```

```
struct Ruta{
    char localidad1[20 + 1];
    char localidad2[20 + 1];
    unsigned distancia;
    unsigned ruta;
};
```

```
FILE *f = fopen("C:\\micarpeta\\rutas", "rb");
```

```
unsigned M[LOCS][LOCS];
```

```
/* 2. Codifique el prototipo (no la definición) de la función BuscarLocalidad. Decida los parámetros y justifique su decisión. */
```

```
/* ...
 */
```

```
int BuscarLocalidad(const char *localidades[], const char *localidad);
```

```
int BuscarLocalidad(const char localidades[][20+1], const char localidad[]);
```

```
int BuscarLocalidad(const string localidades[], const string localidad);
```

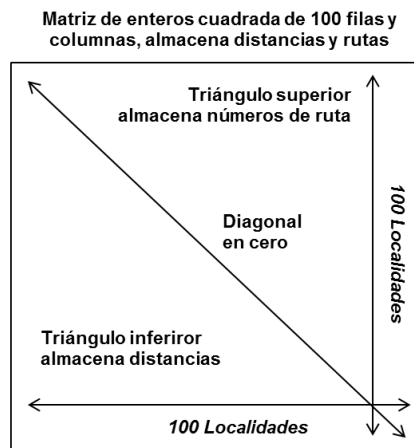
Apellido y nombre: _____ Legajo: _____ Cursó con Prof: _____

Cantidad de hojas entregadas: _____ Nota: _____ Evaluó Prof: _____

- Si luego de la lectura del examen, durante la resolución tiene alguna duda, escriba hipótesis de trabajo, las cuales también serán evaluadas.
- Los puntos que solicitan codificación puede ser respondidos en C, C++, o en Pascal, pero debe indicar el lenguaje utilizado.
- En C y C++ prototipo refiere a la declaración de la función, es decir tipo de dato retornado, nombre de la función, y tipos de los parámetros; en Pascal el concepto prototipo es análogo al encabezado del procedimiento o de la función previa al bloque que lo define.

/*3. Diagrame o codifique la función CargarMatriz

```
void IncializarM(FILE *f, unsigned m[][LOCS], const string v[])
{
```



```
Ruta r;
Int l1,l2;
//Incializar en cero
for(int i
    for(int j
        M[i][j]=0;

while(fread(&r, sizeof(r),1,f){ // puede simplificar
    l1 = BuscarCiudad(v, r.localidad1);
    l2 = BuscarCiudad(v, r.localidad2);

    // Si m[l1][l2] no es triangulo inferior (l1 > l2) intercambio.
    if(l2>l1){
        // intercambio
        int temp = l1;
        l1 = l2;
        l2 = temp;
    }
    //en le siempre está el mayor
    // actualizo la matriz
    M[l1][l2] = r.distancia;
    M[l2][l1] = r.ruta;
}
```

//Nota: aquí no se validan los valores, es decir si hay ciudades inexistentes o si en una cuatroupla se hace referencia a la misma ciudad. Como hipótesis de trabajo suponemos que eso se valida en otro contexto (por ejemplo los datos del archivo fueron validados antes)