

## Estructuras de Datos y Algoritmos

### Práctico de máquina 1 - Año 2026

**Fecha de entrega: Lunes 20 de abril de 2026 hasta las 12 hs.**

Se desea implementar una agenda de prestadores de servicios en la cual se guardaran los siguientes datos: **D.N.I.** del prestador que es único e identifica de forma unívoca toda la información asociada al mismo, nombre y apellido del prestador, servicios prestados, domicilio del prestador, correo electrónico y número de teléfono.

Para almacenar la información planteada se utilizarán las siguientes estructuras evaluándolas con diferentes funciones de pseudo-azar:

- Rebase Separado (**RS1**) con **ABB** en lugar de listas, utilizando la función *hashing1*.
- Rebase Separado (**RS2**) con **ABB** en lugar de listas, utilizando la función *hashing2*.

La aplicación deberá presentar un menú de opciones principal que permita seleccionar las siguientes opciones: **Comparación de estructuras** y **Mostrar Estructura** (una opción de muestra por cada estructura).

La opción **Mostrar Estructura** debe mostrar por pantalla el contenido de la estructura seleccionada y debe mostrar para las **M** posiciones los elementos presentes en cada **ABB** realizando un barrido *in-orden* mostrando para cada nodo la información del prestador y el **D.N.I.** de los nodos hijos; o indicar que la lista está vacía.

**Comparación de Estructuras:** Esta opción debe realizar y mostrar una comparación adecuada de lo que cuesta, en cada una de las estructuras, **realizar altas, bajas y consultas de un prestador dado**. En el análisis debe considerar el peor escenario y el comportamiento esperado en cada caso. Una vez finalizada esta operación, deberá realizar un análisis de los resultados obtenidos y sacar una conclusión de los mismos; dicha conclusión deberá quedar plasmada al principio de su programa principal (donde se encuentra el main) como comentario (**incluir los resultados de la comparación**).

Para el cálculo de los costos la función de costo a utilizar para las evocaciones es celdas consultadas **1 punto por cada celda**, para altas y bajas la función de costos será modificación de punteros **1 punto por cada puntero modificado** y en caso de utilizar política de reemplazo en la operación de baja deberá sumar un punto más por la copia de datos.

Para comparar las estructuras se utilizará una secuencia de operaciones detallada en el archivo de texto "*Operaciones-Prestadores.txt*" que contiene información de prestadores y será provisto por la cátedra (disponible en la *página web de la materia*). Esta secuencia de operaciones se deberá realizar sobre cada una de las estructuras, asegurando que las mismas **no contengan ningún dato antes de comenzar la comparación**. Una vez finalizada la secuencia de operaciones se mostrarán por pantalla los costos obtenidos para cada estructura. Además, una vez terminada la comparación, en las estructuras **deben quedar** los datos resultantes de efectuar las operaciones del archivo para ser visualizados desde la opción mostrar estructura de cada una de ellas.

El archivo de texto "*Operaciones-Prestadores.txt*" contiene una línea con el código de operación (1-Alta, 2-Baja y 3-Evocación) y a continuación los datos de la nupla necesarios para la operación en cada línea (renglón) del mismo. Un ejemplo de esa información se muestra a continuación:



```

1          /*código de la primera operación (Alta)*/
21695670  /*DNI prestador*/
Jorge Icaza /*nombre y apellido prestador*/
Albañil, Plomero /*Servicios*/
4 DE JUNIO 100 /*domicilio prestador*/
jorge@mail.com /*mail prestador*/
2664000000 /*teléfono prestador*/
3          /*código de la segunda operación (Evocación)*/
26545670  /*DNI prestador*/
.
.
.
2          /*código de la n-ésima operación (Baja)*/
33565678 /*DNI prestador*/
Raul Perez /*nombre y apellido prestador*/
Carpintero /*Servicios*/
Rivadavia 1351 /*domicilio prestador*/
raul@mail.com /*mail prestador*/
2664000001 /*teléfono prestador*/

```

### Consideraciones a tener en cuenta:

- Se esperan 210 prestadores.
- La confirmación de la baja de un elemento, en la rutina de baja, debe realizarse de forma automática por código, comparando toda la nupla (en todas las estructuras).
- El factor de carga de cada **RS** será  $\rho = 9.47$ .
- Se utilizará una ranura por balde en cada Rebalse.
- No se utilizarán baldes en la cabecera de los **RS**.
- La política de reemplazo en la baja del **ABB** del **RS1** cuando el nodo tiene dos hijos es el menor de los mayores y el reemplazo deberá realizarse con copia de datos.
- La política de reemplazo en la baja del **ABB** del **RS2** cuando el nodo tiene dos hijos es el mayor de los menores y el reemplazo deberá realizarse con copia de datos.
- Se deben utilizar las siguientes funciones de hashing:

```

int hashing1 (int dni, int M) {
    char x[10];
    int longitud, i;
    int contador=0;
    sprintf(x, '%d', dni);
    longitud=strlen(x);
    for (i=0; i< longitud; i++)
        contador+=((int)x[i]) * (i+1);

    return (contador % M);
}

int hashing2 (int dni, int M) {

    return (dni % M);
}

```

- Los números de D.N.I. son enteros.
- El campo nombre y apellido puede contener un máximo de 80 caracteres en cada caso.



- El campo servicios puede contener un máximo de 120 caracteres.
- El domicilio puede contener un máximo de 80 caracteres.
- El correo electrónico puede contener un máximo de 50 caracteres.
- El número de teléfono puede contener un máximo de 30 caracteres.
- La manipulación de datos **no debe ser sensible a mayúsculas y minúsculas**, esto significa por ejemplo que Juan Fernandez = JUAN FERNANDEZ = JUan fernandez.
- El pseudo-código genérico de los operadores puede verse en el apunte *Operaciones sobre Conjuntos*.
- El programa deberá desarrollarse en Lenguaje C, utilizando como entorno de desarrollo para tal fin **Code::Blocks** (disponible en [www.codeblocks.org](http://www.codeblocks.org)).

### Importante:

- Los grupos deben ser de 2 integrantes.
- Los códigos fuente entregados que no compilen o estén incompletos respecto de la funcionalidad solicitada no serán revisados.
- La entrega del práctico se realiza por medio de la página de la materia y se debe enviar el archivo fuente del programa.
- El nombre del archivo deberá estar conformado de la siguiente manera: ***PnroP-GruponroG*** donde *nroP* es reemplazado por el número de práctico que se entrega y *nroG* por el número del grupo al que pertenece el programa. Por ejemplo, el nombre P1-Grupo22.c corresponde al práctico de máquina 1 enviado por el grupo 22. **Los programas cuyos nombres no respeten estas reglas de conformación no serán aceptados.**