UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Base de Datos II

Profesor Heider Sanchez

2025 - 2

Laboratorio 1: Registros de Longitud Fija y Variable

P1 (10 pts): Registros de Longitud Fija

Dada la siguiente estructura del registro de longitud fija:

```
Registro Alumno:
    Atributos:
        codigo (Cadena[5])       # Tamaño: 5 caracteres
        nombre (Cadena[11])      # Tamaño: 11 caracteres
        apellidos (Cadena[20])   # Tamaño: 20 caracteres
        carrera (Cadena[15])     # Tamaño: 15 caracteres
        ciclo (Entero)           # Entero para el ciclo
        mensualidad (Decimal)    # Decimal para la mensualidad
```

Requisitos:

Se le pide implementar una clase llamada `FixedRecord` que encapsule las operaciones de manipulación de archivo binario con dos estrategias de eliminación:

1. **El constructor** debe recibir el nombre del archivo y el modo de eliminación:
    - **MOVE THE LAST**: mueve el último registro a la posición del registro eliminado.
    - **FREE LIST**: mantiene una lista de espacios libres para ser usados en nuevas inserciones.
    - Puede separar la implementación en dos clases, una para cada modo de eliminación.
2. **Implementar las siguientes funciones:**
    - `load()` : devuelve todos los registros válidos del archivo.
    - `add(record)` : agrega un nuevo registro al archivo O(1). Debe considerar los espacios libres.
    - `readRecord(pos)` : obtiene el registro de la posición "pos" O(1). Debe validar si el registro ha sido eliminado.
    - `remove(pos)` : elimina el registro de la posición "pos" O(1).
3. **Realizar pruebas funcionales** de cada método (`P1.py`).

P2 (10 pts):

Dada la siguiente estructura del registro de longitud variable:

```
Registro Matricula:
    Atributos:
        codigo (Cadena*)        # Tamaño variable
        ciclo (Entero)          # Entero para el ciclo
        mensualidad (Decimal)   # Decimal para la mensualidad
        observaciones (Cadena*) # Tamaño variable
```

Requisitos:

Se le pide implementar un programa para leer y escribir registros de longitud variable en un archivo binario usando el tamaño del dato como separador:

1. **Manejar un archivo adicional** (**metadata**) para indicar la posición inicial de cada registro.
    - Evaluar si es necesario guardar también el tamaño del registro.
2. **Implementar las siguientes funciones:**
    - `load()` : devuelve todos los registros del archivo.
    - `add(record)` : agrega un nuevo registro al archivo O(1).
    - `readRecord(pos)` : obtiene el registro de la posición "pos" O(1).
    - `remove(pos)` : elimina el registro de la posición "pos" (proponer una estrategia eficiente).
3. **Realizar las pruebas funcionales** de cada método (`P2.cpp`).

Entregable

Archivos a entregar:

- P1.py - Ejercicio 1 resuelto en Python (módulo struct)
- **P2.cpp** - Ejercicio 2 resuelto en C++ 17
- **Archivos de datos** utilizados durante las pruebas

Especificaciones de calidad:

- **Pruebas funcionales completas** para cada método implementado
- **Compatibilidad de lectura** - Los datos deben poder leerse correctamente en cualquier ejecución del programa
- **Documentación del código** - Comentarios explicativos en secciones importantes, especialmente en la implementación de las técnicas aplicadas
- **Compilación y ejecución exitosa** - Ambos programas deben compilar y ejecutarse sin errores