



Nombre y Apellido:

Examen Final

Considere las siguientes definiciones:

```
typedef struct {
    void *dato;
} CasillaHash;

struct _TablaHash {
    CasillaHash *elems;
    unsigned numElems;
    unsigned capacidad;
    FuncionCopiadora copia;
    FuncionComparadora comp;
    FuncionDestructora destr;
    FuncionHash hash;
};

typedef struct _TablaHash *TablaHash;
```

Puede utilizar la plantilla provista para tabla hash.

1. Implemente

```
void tablahash_redimensionar(TablaHash)
```

de manera que duplique la capacidad de la tabla y *rehashes* los elementos.

2. Reimplemente `tablahash_insertar` de forma que

- las colisiones se manejen con direccionamiento abierto y
- se redimensione la tabla al doble de su tamaño cuando el factor de carga supere 0.75.

Nota: Analice si es necesario modificar la estructura de la casilla y/o tabla Hash.

3. Reimplemente `tablahash_eliminar` con la técnica que marca el casillero a eliminar como eliminado.

Nota: Analice si es necesario modificar la estructura de la casilla y/o tabla Hash.

4. Reimplemente `tablahash_buscar` para que tenga en cuenta casillas eliminadas.

5. Considere la siguiente estructura:

```
typedef struct {
    char *nombre, *direccion, *dni;
    int edad;
} Persona;
```

Implemente un `main` que cree una `TablaHash` de estructuras `Persona`, donde la clave es el DNI de la persona. Inserte 2 personas en la tabla y luego busque una para verificar que se haya insertado correctamente. Finalmente, elimine una de las personas insertadas y búsqela para verificar que ya no esté presente en la tabla.