

Arowwai Industries



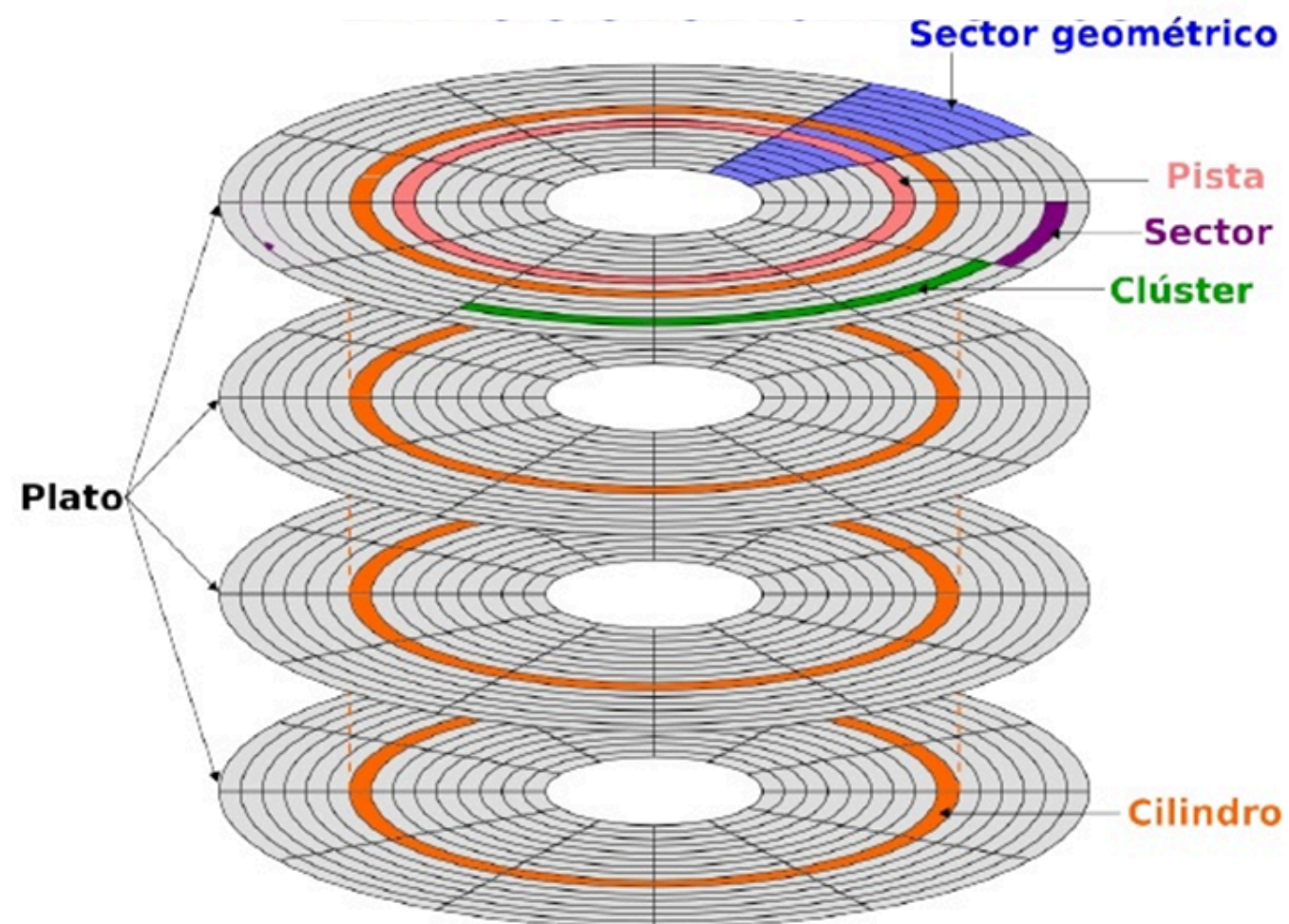
SISTEMA GESTOR DE BASE DE DATOS



JOSE ALEJANDRO MACHACA MUÑIZ



DISCO



15

DiscoDuroSimulado

Plato_0

> Superficie_0

> Superficie_1

Plato_1

> Superficie_0

> Superficie_1

Plato_2

Plato_3

Plato_4

Plato_5



INSERCIÓN SECUENCIAL

```
void escribirRegistro(const string& registro, int pesoRegistro, const string& nombreArchivo) {
    if (pesoRegistro > tamanoSector) {
        cerr << "Registro excede tamaño de sector\n";
        return;
    }

    // Verificar si el disco ya está lleno
    if (espacioActualDisco + pesoRegistro > espacioTotalDisco) {
        cerr << "Disco lleno. No se puede escribir el registro.\n";
        return;
    }

    if (bloque == nullptr) {
        bloque = new Bloque(bloqueActual, nombreArchivo, pesoRegistro, tamanoSector);
    }

    while (true) {
        string pathSector = construirPathSector(plato, superficie, pista, sector);
        string nombreArchivoSector = leerPrimeraLineaArchivo(pathSector);

        if (nombreArchivoSector == "SIN_NOMBRE_AUN" || nombreArchivoSector == nombreArchivo) {
            if (nombreArchivoSector == "SIN_NOMBRE_AUN") {
                actualizarNombreArchivoSector(pathSector, nombreArchivo);
            }

            if (espacioUsadoEnBloque + pesoRegistro > 4 * tamanoSector) {
                delete bloque;
                bloque = new Bloque(++bloqueActual, nombreArchivo, pesoRegistro, tamanoSector);
                espacioUsadoEnBloque = 0;
                sectoresUsadosEnBloque = 0;
                espacioUsadoEnSector = 0;
            }
        }
    }
}
```

```
    if (espacioUsadoEnSector + pesoRegistro > tamanoSector) {
        actualizarPesoSector(espacioUsadoEnSector);
        avanzarSector();
        espacioUsadoEnSector = 0;
        continue;
    }

    ofstream archivo(pathSector, ios::app);
    archivo << registro << endl;
    archivo.close();

    registrarSectorEnBloque(bloqueActual, pathSector);
    bloque->registrarRegistroCompleto(registro, plato, superficie, pista, sector);

    espacioUsadoEnSector += pesoRegistro;
    espacioUsadoEnBloque += pesoRegistro;
    espacioActualDisco += pesoRegistro;

    if (espacioUsadoEnSector == pesoRegistro) {
        sectoresUsadosEnBloque++;
    }

    actualizarPesoSector(espacioUsadoEnSector);
    break;
} else {
    avanzarSector();
    espacioUsadoEnSector = 0;
}
```

CARACTERISTICAS DEL DISCO



```
Seleccione una opcion: 4
Capacidad total del disco: 393216 bytes
Espacio ocupado: 0 bytes
Espacio libre: 393216 bytes
Tamaño de sector 1024 bytes
Espacio por bloque 4096 bytes
```

```
void mostrarEspacioDisco() {
    actualizarEspacioDisco();
    cout << "Capacidad total del disco: " << espacioTotalDisco << " bytes" << endl;
    cout << "Espacio ocupado: " << espacioActualDisco << " bytes" << endl;
    cout << "Espacio libre: " << espacioDisponibleDisco << " bytes" << endl;
    cout << "Tamaño de sector " << tamañoSector << " bytes" << endl;
    cout << "Espacio por bloque " << tamañoSector * 4 << " bytes" << endl;
}
```

ARBOL DEL DISCO

```
void mostrarArbol() {  
    for (int p = 0; p < numPlatos; ++p) {  
        cout << "Plato " << p << endl;  
        for (int s = 0; s < 2; ++s) { // 2 superficies por plato  
            cout << " Superficie " << s << endl;  
            for (int pi = 0; pi < pistasPorSuperficie; ++pi) {  
                cout << " Pista " << pi << endl;  
                for (int se = 0; se < sectoresPorPista; ++se) {  
                    cout << " Sector " << se << endl;  
                }  
            }  
        }  
    }  
}
```

```
07-08-2021  
Seleccione una opcion: 3  
Plato 0  
Superficie 0  
Pista 0  
Sector 0  
Sector 1  
Sector 2  
Sector 3  
Sector 4  
Sector 5  
Sector 6  
Sector 7  
Pista 1  
Sector 0  
Sector 1  
Sector 2  
Sector 3  
Sector 4  
Sector 5  
Sector 6  
Sector 7  
Pista 2  
Sector 0  
Sector 1  
Sector 2  
Sector 3  
Sector 4  
Sector 5  
Sector 6  
Sector 7  
Pista 3  
Sector 0  
Sector 1  
Sector 2  
Sector 3
```

CREACION DEL DISCO

```
void crearEstructura() {
    crearDirectorio(rootDir);
    for (int p = 0; p < numPlatos; ++p) {
        string dirPlato = rootDir + "/Plato_" + to_string(p);
        crearDirectorio(dirPlato);
        for (int s = 0; s < 2; ++s) {
            string dirSuperficie = dirPlato + "/Superficie_" + to_string(s);
            crearDirectorio(dirSuperficie);
            for (int t = 0; t < pistasPorSuperficie; ++t) {
                string dirPista = dirSuperficie + "/Pista_" + to_string(t);
                crearDirectorio(dirPista);
                for (int sec = 0; sec < sectoresPorPista; ++sec) {
                    string rutaSector = dirPista + "/Sector_" + to_string(sec) + ".txt";
                    ofstream archivo(rutaSector);
                    archivo << "SIN_NOMBRE_AUN\n";           // línea 1: nombre archivo pendiente
                    archivo << tamanoSector << "\n";         // línea 2: tamaño total sector
                    archivo << 0 << "\n";                     // línea 3: peso usado
                    archivo.close();
                }
            }
        }
    }
}
```

LONGITUD FIJA



LONGITUD FIJA

INSERTAR 1 REGISTRO

```
void insertarRegistro() {
    string archivo;
    cout << "Ingrese el nombre del archivo (con .txt): ";
    cin >> archivo;

    string atributos[20];
    int cantidad = 0;
    if (!obtenerAtributos(archivo, atributos, cantidad)) {
        cerr << "❌ Atributos no encontrados para " << archivo << endl;
        return;
    }

    cin.ignore();
    string registro = "";
    for (int i = 0; i < cantidad; ++i) {
        string valor;
        cout << "Ingrese " << atributos[i] << ": ";
        getline(cin, valor);
        if (i > 0) registro += "#";
        registro += valor;
    }

    int pesoRegistro = 0;
    int tmpIndex = 0;
    char rutaBloqueTmp[100];
    while (true) {
        sprintf(rutaBloqueTmp, "bloques/bloque%d.txt", tmpIndex);
        ifstream in(rutaBloqueTmp);
        if (!in.is_open()) break;

        string cab;
        getline(in, cab);
        in.close();

        char nombreArchivo[50];
        int lineaInsertar, dummy, peso, numRegs, tamActual, tamTotal;
        sscanf(cab.c_str(), "%[^#]#%d#%d#%d#%d#%d#%d",
```

```
        nombreArchivo, &lineaInsertar, &dummy, &peso, &numRegs, &tamActual, &tamTotal);

        if (archivo == nombreArchivo) {
            pesoRegistro = peso;
            break;
        }
        tmpIndex++;
    }

    if (pesoRegistro == 0) {
        cerr << "❌ No se encontró peso de registro para " << archivo << endl;
        return;
    }

    int bloqueIndexSector = -1;
    string direccionFisica;

    if (!insertarEnSectorEnBloquesArchivo(archivo, registro, pesoRegistro, bloqueIndexSector, direccionFisica)) {
        cerr << "No se pudo insertar registro en sector\n";
        return;
    }

    string registroConDireccion = registro + direccionFisica;

    int bloqueIndexBloque = -1;
    if (!insertarRegistroEnBloque(archivo, registroConDireccion, pesoRegistro, bloqueIndexBloque)) {
        cerr << "No se pudo insertar registro en bloque\n";
        return;
    }

    cout << "✅ Registro insertado correctamente.\n";
```


LONGITUD FIJA



INSERTAR ARCHIVO.TXT

```
EMA-GESTOR-DE-BASE-DE-DATOS-2025- > output > bloques > ≡ bloque0.txt
1  titanic.txt#52#-1#65#50#3250#4096
2  1#0#3#"Braund, Mr. Owen Harris"#male#22#1#0#A/5 21171#7.25##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
3  2#1#1#"Cumings, Mrs. John Bradley (Florence Briggs Thayer)"#female#38#1#0#PC 17599#712.833#C85#C | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
4  3#1#3#"Heikkinen, Miss. Laina"#female#26#0#0#STON/O2. 3101282#7.925##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
5  4#1#1#"Futrelle, Mrs. Jacques Heath (Lily May Peel)"#female#35#1#0#113803#53.1#C123#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
6  5#0#3#"Allen, Mr. William Henry"#male#35#0#0#373450#08.05##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
7  6#0#3#"Moran, Mr. James"#male##0#0#330877#8.4583##Q | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
8  7#0#1#"McCarthy, Mr. Timothy J"#male#54#0#0#17463#518.625#E46#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
9  8#0#3#"Palsson, Master. Gosta Leonard"#male#2#3#1#349909#21.075##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
10 9#1#3#"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)"#female#27#0#2#347742#11.1333##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
11 10#1#2#"Nasser, Mrs. Nicholas (Adele Achem)"#female#14#1#0#237736#300.708##C | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
12 11#1#3#"Sandstrom, Miss. Marguerite Rut"#female#4#1#1#PP 9549#16.7#G6#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
13 12#1#1#"Bonnell, Miss. Elizabeth"#female#58#0#0#113783#26.55#C103#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
14 13#0#3#"Saunderscock, Mr. William Henry"#male#20#0#0#A/5. 2151#08.05##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
15 14#0#3#"Andersson, Mr. Anders Johan"#male#39#1#5#347082#31.275##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
16 15#0#3#"Vestrom, Miss. Hulda Amanda Adolfina"#female#14#0#0#350406#7.8542##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
```

```
EMA-GESTOR-DE-BASE-DE-DATOS-2025- / output / DiscoDatosSimulado / Plato_0 / Superficie_0 / Pista_0 / ≡ Sector_0.txt
titanic.txt
1024
975
1#0#3#"Braund, Mr. Owen Harris"#male#22#1#0#A/5 21171#7.25##S
2#1#1#"Cumings, Mrs. John Bradley (Florence Briggs Thayer)"#female#38#1#0#PC 17599#712.833#C85
3#1#3#"Heikkinen, Miss. Laina"#female#26#0#0#STON/O2. 3101282#7.925##S
4#1#1#"Futrelle, Mrs. Jacques Heath (Lily May Peel)"#female#35#1#0#113803#53.1#C123#S
5#0#3#"Allen, Mr. William Henry"#male#35#0#0#373450#08.05##S
6#0#3#"Moran, Mr. James"#male##0#0#330877#8.4583##Q
7#0#1#"McCarthy, Mr. Timothy J"#male#54#0#0#17463#518.625#E46#S
8#0#3#"Palsson, Master. Gosta Leonard"#male#2#3#1#349909#21.075##S
9#1#3#"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)"#female#27#0#2#347742#11.1333##S
10#1#2#"Nasser, Mrs. Nicholas (Adele Achem)"#female#14#1#0#237736#300.708##C
11#1#3#"Sandstrom, Miss. Marguerite Rut"#female#4#1#1#PP 9549#16.7#G6#S
12#1#1#"Bonnell, Miss. Elizabeth"#female#58#0#0#113783#26.55#C103#S
13#0#3#"Saunderscock, Mr. William Henry"#male#20#0#0#A/5. 2151#08.05##S
14#0#3#"Andersson, Mr. Anders Johan"#male#39#1#5#347082#31.275##S
15#0#3#"Vestrom, Miss. Hulda Amanda Adolfina"#female#14#0#0#350406#7.8542##S
```

LONGITUD FIJA

INSERTAR 1 REGISTRO



```
titanic.txt#52#-1#65#50#3250#4096
1#0#3#"Braund, Mr. Owen Harris"#male#22#1#0#A/5 21171#7.25##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
2#1#1#"Cumings, Mrs. John Bradley [Florence Briggs Thayer]"#female#38#1#0#PC 17599#712.833#C85#C | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
3#1#1#"Bonn, Mr. Louis Alexander"##male#28#1#0#S.C./A.4. 21013#7.25##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
49#0#3#"Samaan, Mr. Youssef"#male##2#0#2662#216.792##C | Pos: Plato 0, Superficie 0, Pista 0, Sector 3
51 50#0#3#"Arnold-Franchi, Mrs. Josef (Josefine Franchi)"#female#18#1#0#349237#17.8##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 3
52
```

titanic.txt#53#-1#65#51#3315#4096

```
52 1000#10#50#Q#Q#Q#Q#Q#Q#Q#Q#Q | Pos: Plato 0, Superficie 0, Pista 0, Sector 3
53
```

SISTEMA-GESTOR-DE-BASE-DE-DATOS-2025- > output > DiscoDuroSimulado > Plato_0 > Superficie_0 > Pista_0 > Sector_3.txt

```
1 titanic.txt
2 1024
3 390
4 46#0#3#"Rogers, Mr. William John"#male##0#0#S.C./A.4. 23567#08.05##S
5 47#0#3#"Lennon, Mr. Denis"#male##1#0#370371#15.5##Q
6 48#1#3#"O'Driscoll, Miss. Bridget"#female##0#0#14311#7.75##Q
7 49#0#3#"Samaan, Mr. Youssef"#male##2#0#2662#216.792##C
8 50#0#3#"Arnold-Franchi, Mrs. Josef (Josefine Franchi)"#female#18#1#0#349237#17.8##S
9 1000#10#50#Q#Q#Q#Q#Q#Q#Q#Q#Q
10
```

LONGITUD FIJA INSERTAR CON EL SECTOR LLENO



EN ESTE CASO VAMOS A VER QUE SECTORES TIENE UN BLOQUE Y VAMOS
MANIPULAR EL ESPACIO DE UN SECTOR PARA HACER LA SIMULACION

BLOQUEO.TXT

```
DiscoDuroSimulado/Plato_0/Superficie_0/Pista_1/Sector_1.txt  
DiscoDuroSimulado/Plato_0/Superficie_0/Pista_1/Sector_2.txt  
DiscoDuroSimulado/Plato_0/Superficie_0/Pista_2/Sector_0.txt  
DiscoDuroSimulado/Plato_0/Superficie_0/Pista_2/Sector_1.txt  
DiscoDuroSimulado/Plato_0/Superficie_0/Pista_2/Sector_2.txt
```

LONGITUD FIJA INSERTAR CON EL SECTOR LLENO

```
housing.txt
1024
1008
8680000#755#3#2#1#yes#yes#yes#no#yes#2#no#unfurnished
8645000#8050#3#1#1#yes#yes#yes#no#yes#1#no#furnished
8645000#4560#3#2#2#yes#yes#yes#no#yes#1#no#furnished
8575000#8800#3#2#2#yes#no#no#no#yes#2#no#furnished
8540000#6540#4#2#2#yes#yes#yes#no#yes#2#yes#furnished
8463000#6000#3#2#4#yes#yes#yes#no#yes#0#yes#semi-furnished
8400000#8875#3#1#1#yes#no#no#no#no#1#no#semi-furnished
8400000#7950#5#2#2#yes#no#yes#yes#no#2#no#unfurnished
8400000#5500#4#2#2#yes#no#yes#no#yes#1#yes#semi-furnished
8400000#7475#3#2#4#yes#no#no#no#yes#2#no#unfurnished
8400000#7000#3#1#4#yes#no#no#no#yes#2#no#semi-furnished
8295000#4880#4#2#2#yes#no#no#no#yes#1#yes#furnished
8190000#5960#3#3#2#yes#yes#yes#no#no#1#no#unfurnished
8120000#6840#5#1#2#yes#yes#yes#no#yes#1#no#furnished
8080940#7000#3#2#4#yes#no#no#no#yes#2#no#furnished
8043000#7482#3#2#3#yes#no#no#yes#no#1#yes#furnished
7980000#9000#4#2#4#yes#no#no#no#yes#2#no#furnished
7962500#6000#3#1#4#yes#yes#no#no#yes#2#no#unfurnished
7910000#6000#4#2#4#yes#no#no#no#yes#1#no#semi-furnished
7875000#6550#3#1#2#yes#no#yes#no#yes#0#yes#furnished
7840000#6360#3#2#4#yes#no#no#no#yes#0#yes#furnished
```

vamos a manipular su peso manualmente

LONGITUD FIJA INSERTAR CON EL SECTOR LLENO

```
MAQUETOR-DE-BASE-DE-DATOS-2023 / output / DISC-DATOS-simulado / Flaco_0 / superficie_  
housing.txt  
1024  
970  
8680000#7155#3#2#1#yes#yes#yes#no#yes#2#no#unfurnished  
8645000#8050#3#1#1#yes#yes#yes#no#yes#1#no#furnished  
8645000#4560#3#2#2#yes#yes#yes#no#yes#1#no#furnished  
8575000#8800#3#2#2#yes#no#no#no#yes#2#no#furnished  
8540000#6540#4#2#2#yes#yes#yes#no#yes#2#yes#furnished  
8463000#6000#3#2#4#yes#yes#yes#no#yes#0#yes#semi-furnished  
8400000#8875#3#1#1#yes#no#no#no#no#1#no#semi-furnished  
8400000#7950#5#2#2#yes#no#yes#yes#no#2#no#unfurnished  
8400000#5500#4#2#2#yes#no#yes#no#yes#1#yes#semi-furnished  
8400000#7475#3#2#4#yes#no#no#no#yes#2#no#unfurnished  
8400000#7000#3#1#4#yes#no#no#no#yes#2#no#semi-furnished  
8295000#4880#4#2#2#yes#no#no#no#yes#1#yes#furnished  
8190000#5960#3#3#2#yes#yes#yes#no#no#1#no#unfurnished  
8120000#6840#5#1#2#yes#yes#yes#no#yes#1#no#furnished  
8080940#7000#3#2#4#yes#no#no#no#yes#2#no#furnished  
8043000#7482#3#2#3#yes#no#no#yes#no#1#yes#furnished  
7980000#9000#4#2#4#yes#no#no#no#yes#2#no#furnished  
7962500#6000#3#1#4#yes#yes#no#no#yes#2#no#unfurnished  
7910000#6000#4#2#4#yes#no#no#no#yes#1#no#semi-furnished  
7875000#6550#3#1#2#yes#no#yes#no#yes#0#yes#furnished  
7840000#6360#3#2#4#yes#no#no#no#yes#0#yes#furnished
```

PRIMERO LE PEDIRA AL USUARIO QUE ARCHIVO TXT DESEA INSERTAR, LUEGO EL SISTEMA BUSCARA EN LOS BLOQUES QUE TIPO DE ARCHIVO TIENE, DESPUES BUSCARA LOS SECTORES QUE PERTENESCAN A DICHO BLOQUE Y BUSCARA UN ESPACIO DISPONIBLE EN LOS SECTORES DE DICHO BLOQUE, SI NO ENCUENTRA ESPACIO BUSCARA EN OTRO BLOQUE Y SUS SECTORES ASI SUCESIVAMENTE HASTA ENCONTRAR UN ESPACIO DISPONIBLE.

SECTORES OCUPADOS

Total sectores en este bloque: 4

[Bloque: bloque1.txt]

Sector 1 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_0/Pista_1/Sector_1.txt

Sector 2 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_0/Pista_1/Sector_2.txt

Sector 3 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_0/Pista_2/Sector_0.txt

Sector 4 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_0/Pista_2/Sector_1.txt

Sector 5 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_0/Pista_2/Sector_2.txt

Total sectores en este bloque: 5

[Bloque: bloque2.txt]

Sector 1 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_0/Pista_2/Sector_2.txt

Sector 2 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_1/Pista_0/Sector_0.txt

Sector 3 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_1/Pista_0/Sector_1.txt

Sector 4 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_1/Pista_0/Sector_2.txt

Sector 5 -> Dirección: DiscoDuroSimulado/Plato_0/Superficie_1/Pista_1/Sector_0.txt

Total sectores en este bloque: 5

LONGITUD FIJA

ELIMINACION DE

REGISTROS



```
bool eliminarRegistro(int bloqueIndex, const string& idEliminar) {
    string rutaSectores = "bloques_sectores/bloque" + to_string(bloqueIndex) + ".txt";
    ifstream listaSectores(rutaSectores);
    if (!listaSectores.is_open()) {
        cerr << "No se encontró archivo de sectores para bloque " << bloqueIndex << endl;
        return false;
    }
}
```

```
string cabeceraBloque;
if (!leerCabeceraBloque(bloqueIndex, cabeceraBloque)) return false;
```

```
// Parsear cabecera para obtener peso actual y número de registros y peso registro (cuarto parám
vector<string> partes;
size_t pos = 0, prev = 0;
while ((pos = cabeceraBloque.find('#', prev)) != string::npos) {
    partes.push_back(cabeceraBloque.substr(prev, pos - prev));
    prev = pos + 1;
}
```

```
partes.push_back(cabeceraBloque.substr(prev));
```

```
if (partes.size() < 7) {
    cerr << "Formato cabecera inválido" << endl;
    return false;
}
```

```
int pesoActualBloque = stoi(partes[5]);
int numRegsBloque = stoi(partes[4]);
int pesoRegistro = stoi(partes[3]); // cuarto parámetro, índice 3
```

```
int lineaEnBloque = -1;
string rutaBloque = "bloques/bloque" + to_string(bloqueIndex) + ".txt";
ifstream bloqueIn(rutaBloque);
if (!bloqueIn.is_open()) {
    cerr << "No se pudo abrir bloque " << bloqueIndex << endl;
    return false;
}
```

```
1 // Buscar registro en bloque (ignorando cabecera)
2 for (size_t i = 1; i < lineasBloque.size(); i++) {
3     string reg = lineasBloque[i];
4     size_t posSep = reg.find('#');
5     if (posSep != string::npos) {
6         string id = reg.substr(0, posSep);
7         if (id == idEliminar) {
8             lineaEnBloque = (int)i;
9             break;
10        }
11    }
12 }
13 if (lineaEnBloque == -1) {
14     cout << "No se encontró registro con ID " << idEliminar << " en bloque " << bloqueIndex << endl;
15     return false;
16 }
17
```

```
8 string sectorPath;
9 int lineaEnSector = -1;
```

```
1 crearCarpetaSiNoExiste("registros_eliminados");
2 string pathEliminados = "registros_eliminados/bloque" + to_string(bloqueIndex) + ".txt";
3 ofstream outEliminados(pathEliminados, ios::app);
4 if (!outEliminados.is_open()) {
5     cerr << "No se pudo abrir archivo para registros eliminados: " << pathEliminados << endl;
6     return false;
7 }
8
```

```
9 bool eliminado = false;
```

```
1 while (getline(listaSectores, sectorPath)) {
2     ifstream sectorIn(sectorPath);
3     if (!sectorIn.is_open()) {
4         cerr << "No se pudo abrir sector: " << sectorPath << endl;
5         continue;
6     }
7 }
```

LONGITUD FIJA

ELIMINACION DE

REGISTROS



```
        return false;
    }
    sectorOut << nombreArchivoSector << "\n" << tamTotalStr << "\n" << tamActualStr << "\n";
    for (auto& r : registrosSector) {
        sectorOut << r << "\n";
    }
    sectorOut.close();

    // Vaciar línea en el bloque
    lineasBloque[lineaEnBloque] = "";

    // Reescribir bloque
    ofstream bloqueOut(rutaBloque);
    if (!bloqueOut.is_open()) {
        cerr << "No se pudo abrir bloque para escritura: " << rutaBloque << endl;
        return false;
    }
    for (const string& l : lineasBloque) {
        bloqueOut << l << "\n";
    }
    bloqueOut.close();

    // Actualizar cabecera del bloque
    int nuevoPesoBloque = pesoActualBloque - pesoRegistro;
    int nuevoNumRegs = numRegsBloque - 1;
    if (!actualizarCabeceraBloque(bloqueIndex, nuevoPesoBloque, nuevoNumRegs)) return false;

    // Guardar info de eliminación
    outEliminados << idEliminar << "#" << lineaEnBloque + 1 << "#" << lineaEnSector << "\n";

    eliminado = true;
    break;
}
}
if (eliminado) break;
```

```
    }

    outEliminados.close();

    if (!eliminado) {
        cout << "No se encontró registro con ID " << idEliminar << " en sectores del bloque " << bloqueIndex << endl;
        return false;
    }

    cout << "Registro con ID " << idEliminar << " eliminado correctamente." << endl;
    return true;
```

LONGITUD FIJA

ELIMINACION DE

REGISTROS



```
titanic.txt#52#-1#65#50#3250#4096
1#0#3#"Braund, Mr. Owen Harris"#male#22#1#0#A/5 21171#7.25##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
2#1#1#"Cumings, Mrs. John Bradley (Florence Briggs Thayer)"#female#38#1#0#PC 17599#712.833#C85#C | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
3#1#3#"Heikkinen, Miss. Laina"#female#26#0#0#STON/02. 3101282#7.925##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
4#1#1#"Futrelle, Mrs. Jacques Heath (Lily May Peel)"#female#35#1#0#113803#53.1#C123#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
5#0#3#"Allen, Mr. William Henry"#male#35#0#0#373450#08.05##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
6#0#3#"Moran, Mr. James"#male##0#0#330877#8.4583##Q | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
7#0#1#"McCarthy, Mr. Timothy J"#male#54#0#0#17463#518.625#E46#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
8#0#3#"Palsson, Master. Gosta Leonard"#male#2#3#1#349909#21.075##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
```

CUANDO SE ELIMINA UN REGISTRO EL EMPTY CAMBIA A 0

```
1  titanic.txt#52#0#65#49#3185#4096
2  1#0#3#"Braund, Mr. Owen Harris"#male#22#1#0#A/5 21171#7.25##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
3
4  3#1#3#"Heikkinen, Miss. Laina"#female#26#0#0#STON/02. 3101282#7.925##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
5  4#1#1#"Futrelle, Mrs. Jacques Heath (Lily May Peel)"#female#35#1#0#113803#53.1#C123#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
6  5#0#3#"Allen, Mr. William Henry"#male#35#0#0#373450#08.05##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0
```

LONGITUD FIJA

ELIMINACION DE

REGISTROS



```
MA-GESTOR-DE-BASE-DE-DATOS-2025- > output > DiscoDuroSimulado > Plato_0 > Superficie_0 > Pista_0 > ≡ Sector_0.txt
```

```
titanic.txt
```

```
1024
```

```
910
```

```
1#0#3#"Braund, Mr. Owen Harris"#male#22#1#0#A/5 21171#7.25##S
```

```
3#1#3#"Heikkinen, Miss. Laina"#female#26#0#0#STON/02. 3101282#7.925##S
```

```
4#1#1#"Futrelle, Mrs. Jacques Heath (Lily May Peel)"#female#35#1#0#113803#53.1#C123#S
```

```
5#0#3#"Allen, Mr. William Henry"#male#35#0#0#373450#08.05##S
```

```
6#0#3#"Moran, Mr. James"#male##0#0#330877#8.4583##Q
```

```
7#0#1#"McCarthy, Mr. Timothy J"#male#54#0#0#17463#518.625#E46#S
```

```
8#0#3#"Palsson, Master. Gosta Leonard"#male#2#3#1#349909#21.075##S
```

```
9#1#3#"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)"#female#27#0#2#347742#11.1333##S
```

```
10#1#2#"Nasser, Mrs. Nicholas (Adele Achem)"#female#14#1#0#237736#300.708##C
```

```
11#1#3#"Sandstrom, Miss. Marguerite Rut"#female#4#1#1#PP 9549#16.7#G6#S
```

```
12#1#1#"Bonnell, Miss. Elizabeth"#female#58#0#0#113783#26.55#C103#S
```

```
13#0#3#"Saunderscock, Mr. William Henry"#male#20#0#0#A/5. 2151#08.05##S
```

```
14#0#3#"Andersson, Mr. Anders Johan"#male#39#1#5#347082#31.275##S
```

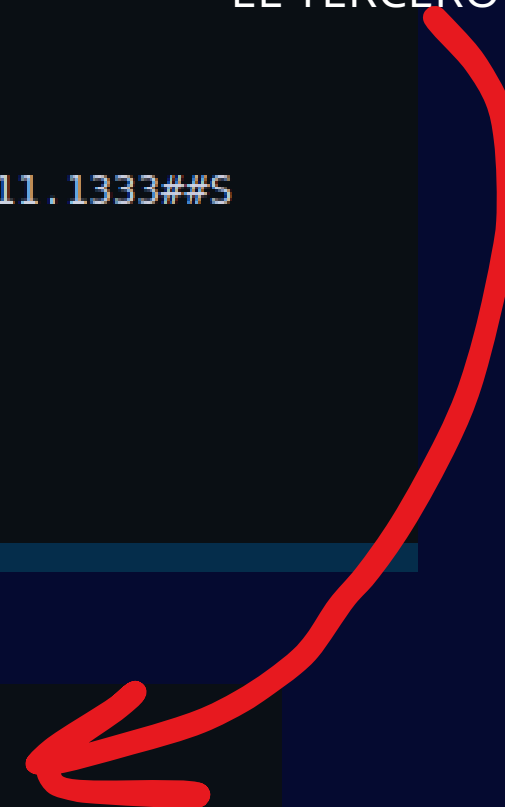
```
15#0#3#"Vestrom, Miss. Hulda Amanda Adolfina"#female#14#0#0#350406#7.8542##S
```

EL PROGRAMA LE PEDIRA AL USUARIO QUE REGISTRO DEL BLOQUEX DESEA ELIMINAR, EL PROGRAMA ELIMINARA EL REGISTRO DEL BLOQUE Y DEL SECTOR , ADEMAS GUARDARA EN QUE LINEA DEL BLOQUE Y DEL SECTOR LO ELIMINA PARA USARLO EN CASO DE INSERCION.

EL SEGUNDO PARAMETRO INDICARA LA LINEA DEL BLOQUE Y EL TERCERO LA LINEA DEL SECTOR

```
SISTEMA-GESTOR-DE-BASE-DE-DATOS-2025- > output > registros_eliminados > ≡ bloque0.txt
```

```
1 2#3#5
```



11

1 2#3#5

EW0434411 on Mr. William Henry #mal043540404373450400 05#40 1 Bag: 0 Plate: 0 Superficie: 0 Distr: 0 Factor: 0

3#1#3#"Heikkinen, Miss. Laina"#female#26#0#0#STON/02. 3101282#7.925##S

CARACTERISTICAS DE BLOQUE



[BLOQUE 0]

Archivo origen: titanic.txt

Tamaño por registro: 65 bytes

Registros almacenados: 50

Espacio usado: 3250 bytes

Capacidad total: 4096 bytes

Porcentaje ocupado: 79.3457%

[REGISTROS]

1: 1#0#3#"Braund, Mr. Owen Harris"#male#22#1#0#A/5 21171#7.25##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

2: GUSTAV0#Q#Q#Q#Q#Q#Q#Q#Q#Q#Q#Q | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

3: 3#1#3#"Heikkinen, Miss. Laina"#female#26#0#0#STON/02. 3101282#7.925##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

4: 4#1#1#"Futrelle, Mrs. Jacques Heath (Lily May Peel)"#female#35#1#0#113803#53.1#C123#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

5: 5#0#3#"Allen, Mr. William Henry"#male#35#0#0#373450#08.05##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

6: 6#0#3#"Moran, Mr. James"#male##0#0#330877#8.4583##Q | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

7: 7#0#1#"McCarthy, Mr. Timothy J"#male#54#0#0#17463#518.625#E46#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

8: 8#0#3#"Palsson, Master. Gosta Leonard"#male#2#3#1#349909#21.075##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

9: 9#1#3#"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)"#female#27#0#2#347742#11.1333##S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

10: 10#1#2#"Nasser, Mrs. Nicholas (Adele Achem)"#female#14#1#0#237736#300.708##C | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

11: 11#1#3#"Sandstrom, Miss. Marguerite Rut"#female#4#1#1#PP 9549#16.7#G6#S | Pos: Plato 0, Superficie 0, Pista 0, Sector 0

CONSULTAS



0. Salir

Seleccione una opcion: 3

Realizar consulta ejemplo (&SELECT * FROM titanic#) o (&SELECT Name, Age FROM titanic#): &SELECT Name, Age FROM titanic#

Contenido de la tabla 'titanic':

Name	Age
------	-----

"Braund, Mr. Owen Harris"	22
"Cumings, Mrs. John Bradley (Florence Briggs Thayer)"	38
"Heikkinen, Miss. Laina"	26
"Futrelle, Mrs. Jacques Heath (Lily May Peel)"	35
"Allen, Mr. William Henry"	35
"Moran, Mr. James"	
"McCarthy, Mr. Timothy J"	54
"Palsson, Master. Gosta Leonard"	2
"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)"	27
"Nasser, Mrs. Nicholas (Adele Achem)"	14
"Sandstrom, Miss. Marguerite Rut"	4