

Universidad Nacional del Altiplano
Facultad de Ingeniería Estadística e Informática
Docente: Fred Torres Cruz
Autor : Israel Josias Huanacuni Chura

Trabajo Encargado - N° 003

// Ejercicio N° 3

```
#include <iostream>
#include <fstream>
#include <sstream>
#include <string>
#include <iomanip>

int main() {
    std::ifstream inputFile("precios.txt");
    std::ofstream outputFile("precios_soles.txt");

    const double tasa_conversion = 3.85;

    std::string line;
    while (std::getline(inputFile, line)) {
        std::istringstream iss(line);
        std::string producto;
        double precio_dolares;
        char delim;
        std::getline(iss, producto, ',');
        iss >> precio_dolares >> delim;
        double precio_soles = precio_dolares * tasa_conversion;
        outputFile << std::fixed << std::setprecision(2) << producto << ", " << precio_soles
    }

    inputFile.close();
    outputFile.close();

    std::cout << "Los precios han sido convertidos y guardados en 'precios_soles.txt'."
    return 0;
}
```

```
// Ejercicio N° 4

#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <limits>
using namespace std;

struct Registro {
    string fecha;
    double temperatura;
};

int main() {
    ifstream archivoEntrada("temperaturas.txt");
    ofstream archivoSalida("temperaturas_extremas.txt");

    if (!archivoEntrada.is_open()) {
        cerr << "No se pudo abrir el archivo" << endl;
        return 1;
    }

    if (!archivoSalida.is_open()) {
        cerr << "No se pudo abrir el archivo" << endl;
        return 1;
    }

    string linea;
    Registro maxTemp, minTemp;
    maxTemp.temperatura = -numeric_limits<double>::infinity();
    minTemp.temperatura = numeric_limits<double>::infinity();

    while (getline(archivoEntrada, linea)) {
        stringstream ss(linea);
        Registro registro;
        if (getline(ss, registro.fecha, ',') && ss >> registro.temperatura) {
            if (registro.temperatura > maxTemp.temperatura) {
                maxTemp = registro;
            }
            if (registro.temperatura < minTemp.temperatura) {
                minTemp = registro;
            }
        } else {
```

```
        cerr << "Error: formato incorrecto en la línea: " << linea << endl;
    }
}

archivoEntrada.close();

archivoSalida << "Día de temperatura máxima: " << maxTemp.fecha << ", " << maxTemp.t
archivoSalida << "Día de temperatura mínima: " << minTemp.fecha << ", " << minTemp.t

archivoSalida.close();

cout << "Resultados guardados en 'temperaturas_extremas.txt'." << endl;

return 0;
}

// Ejercicio N° 5

#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <limits>
using namespace std;

struct Registro {
    string fecha;
    double venta;
};

int main() {
    ifstream archivoEntrada("ventas_diarias.txt");
    ofstream archivoSalida("resumen_ventas.txt");

    if (!archivoEntrada.is_open()) {
        cerr << "No se pudo abrir el archivo de entrada." << endl;
        return 1;
    }

    if (!archivoSalida.is_open()) {
        cerr << "No se pudo abrir el archivo de salida." << endl;
        return 1;
    }
}
```

```
string linea;
Registro maxVenta, minVenta;
maxVenta.venta = -numeric_limits<double>::infinity(); // Inicializa a un valor muy
minVenta.venta = numeric_limits<double>::infinity();   // Inicializa a un valor muy
double ventaTotal = 0.0;
int contador = 0;

while (getline(archivoEntrada, linea)) {
    stringstream ss(linea);
    Registro registro;
    if (getline(ss, registro.fecha, ',') && ss >> registro.venta) {
        ventaTotal += registro.venta;
        contador++;

        if (registro.venta > maxVenta.venta) {
            maxVenta = registro;
        }
        if (registro.venta < minVenta.venta) {
            minVenta = registro;
        }
    } else {
        cerr << "Error: formato incorrecto en la línea: " << linea << endl;
    }
}

archivoEntrada.close();

double promedioVentas = (contador > 0) ? (ventaTotal / contador) : 0.0;

archivoSalida << "Venta total: " << ventaTotal << endl;
archivoSalida << "Promedio de ventas: " << promedioVentas << endl;
archivoSalida << "Día de mayor venta: " << maxVenta.fecha << ", " << maxVenta.venta
archivoSalida << "Día de menor venta: " << minVenta.fecha << ", " << minVenta.venta

archivoSalida.close();

cout << "Procesamiento completo. Resultados guardados en 'resumen_ventas.txt'." << endl;

return 0;
}
```