**Mezcla Directa**

**1. Introducción**

El algoritmo de mezcla directa es un método de ordenación interna que utiliza el enfoque de "divide y conquista" para ordenar una lista de elementos. En este documento, exploraremos el algoritmo de mezcla directa, su funcionamiento y su implementación en C++.

**2. Algoritmo de Mezcla Directa**

El algoritmo de mezcla directa sigue los siguientes pasos:

1. **División**: Divide la lista en dos sub-listas de tamaño aproximadamente igual.
2. **Ordenación Recursiva**: Ordena cada sub-lista de forma recursiva aplicando el algoritmo de mezcla directa.
3. **Mezcla**: Combina las dos sub-listas ordenadas para obtener la lista ordenada final.

**3. Implementación del Algoritmo de Mezcla Directa**

A continuación, se presenta una implementación del algoritmo de mezcla directa en C++:

#include <iostream>

#include <vector>

**void** **merge**(std::vector<**int**>& arr, **int** inicio, **int** medio, **int** fin) {

**int** n1 = medio - inicio + **1**;

**int** n2 = fin - medio;

std::vector<**int**> L(n1), R(n2);

**for** (**int** i = **0**; i < n1; i++)

L[i] = arr[inicio + i];

**for** (**int** j = **0**; j < n2; j++)

R[j] = arr[medio + **1** + j];

**int** i = **0**, j = **0**, k = inicio;

**while** (i < n1 && j < n2) {

**if** (L[i] <= R[j]) {

arr[k] = L[i];

i++;

} **else** {

arr[k] = R[j];

j++;

}

k++;

}

**while** (i < n1) {

arr[k] = L[i];

i++;

k++;

}

**while** (j < n2) {

arr[k] = R[j];

j++;

k++;

}

}

**void** **mergeSort**(std::vector<**int**>& arr, **int** inicio, **int** fin) {

**if** (inicio < fin) {

**int** medio = inicio + (fin - inicio) / **2**;

mergeSort(arr, inicio, medio);

mergeSort(arr, medio + **1**, fin);

merge(arr, inicio, medio, fin);

}

}

**void** **imprimirArray**(**const** std::vector<**int**>& arr) {

**for** (**int** num : arr) {

std::cout << num << " ";

}

std::cout << std::endl;

}

**int** **main**() {

std::vector<**int**> arr = {**12**, **11**, **13**, **5**, **6**, **7**};

**int** n = arr.size();

std::cout << "Arreglo original:**\n**";

imprimirArray(arr);

mergeSort(arr, **0**, n - **1**);

std::cout << "Arreglo ordenado:**\n**";

imprimirArray(arr);

**return** **0**;

}

**4. Conclusiones**

El algoritmo de mezcla directa ofrece una forma eficiente de ordenar listas de elementos utilizando el enfoque de divide y conquista. Su implementación en C++ proporciona una herramienta poderosa para ordenar datos de manera eficiente.

**5. Referencias**

* Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). The MIT Press.