Report

**Introduction** This program analyzes file sizes in a given directory and categorizes them into bandwidth increments. The purpose is to count the number of files within each bandwidth range and display the results accordingly.

**Function: print\_file\_sizes** **Purpose:** Processes files in a directory and groups them based on size ranges.

**Explanation:**

1. Initializes a map to store file names and their corresponding sizes.
2. Iterates through the given directory using filesystem::directory\_iterator to retrieve file sizes.
3. Determines the largest file size to define the range of bandwidths.
4. Iterates through file size ranges, checking how many files fall within each bandwidth interval.
5. Uses an iterator to safely remove processed files from the map to avoid redundant calculations.
6. Outputs the number of files within each bandwidth range.

**Main Function** **Purpose:** Calls print\_file\_sizes with a specified directory and bandwidth increment.

**Explanation:**

1. Defines a directory path where the file size analysis will be performed.
2. Calls print\_file\_sizes with a bandwidth\_increments value to categorize files.
3. Displays the results of file distribution across different size ranges.

**Conclusion** This program effectively categorizes and counts files within predefined size intervals. By leveraging the C++ filesystem library, it efficiently processes directory contents and provides insights into file size distribution. The use of a map ensures structured storage of file information, and the iterator-based approach optimizes processing by avoiding unnecessary re-iterations.

Source Code

#include <iostream>

#include <string>

#include <vector>

#include <filesystem>

#include <map>

using namespace std;

void print\_file\_sizes(int bandwidth\_increments, string& directory)

{

map<string, int> filename\_size;

int file\_count = 0;

int max\_size = 0;

for (const auto& entry : filesystem::directory\_iterator(directory))

{

int file\_size = filesystem::file\_size(entry.path());

filename\_size[entry.path().string()] = file\_size;

file\_count++;

if (file\_size > max\_size)

{

max\_size = file\_size;

}

}

for (int i = bandwidth\_increments; i < max\_size; i += bandwidth\_increments)

{

int files\_in\_bandwidth = 0;

auto it = filename\_size.begin();

while (it != filename\_size.end()) {

if (it->second < i) {

files\_in\_bandwidth++;

it = filename\_size.erase(it);

}

else {

++it;

}

}

cout << "Number of files in bandwidth " << i - bandwidth\_increments << " to " << i << " is/are: " << files\_in\_bandwidth << endl;

}

}

int main()

{

string directory = "C:\\Users\\salla\\OneDrive\\Desktop\\Operating-Systems\\Task 5\\Test\_dir";

print\_file\_sizes(1024, directory);

return 0;

}