محمد بسيوني رمضان داود

D4

Bitonic Sort

1) **Create a Bitonic Sequence:**

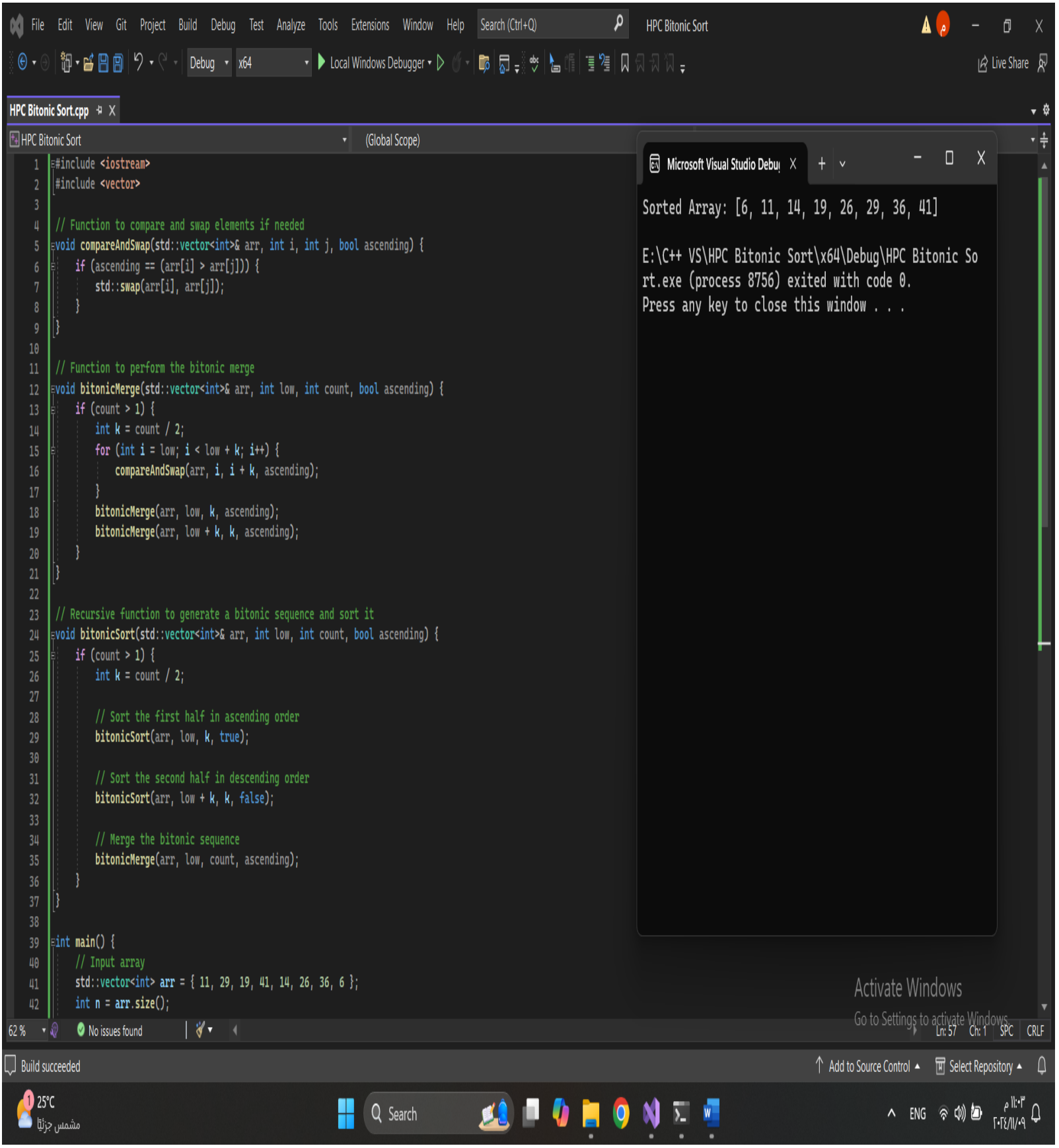
- **Divide the Array into Two Halves:**
  - **First Half: [11, 29, 19, 41] (to be sorted in ascending order).**
  - **Second Half: [14, 26, 36, 6] (to be sorted in descending order).**
- **Sort the First Half in Ascending Order:**
  - **Split [11, 29, 19, 41] into [11, 29] and [19, 41].**
    - **[11, 29] is already in ascending order; no swaps needed.**
    - **[19, 41] is already in ascending order; no swaps needed.**
  - **Merge [11, 29, 19, 41] in ascending order:**
    - **Compare 11 with 19 and 29 with 41. No swaps are needed.**
    - **Resulting First Half: [11, 29, 19, 41]**
- **Sort the Second Half in Descending Order:**
  - **Split [14, 26, 36, 6] into [14, 26] and [36, 6].**

        - **Rearrange [14, 26] to [26, 14] to make it descending.**
        - **Rearrange [36, 6] to [36, 6] (already in descending order).**
    - **Merge [26, 14, 36, 6] in descending order:**
        - **Resulting Second Half: [36, 26, 14, 6]**
- **Resulting Bitonic Sequence:**
    - **After sorting each half, we have [11, 29, 19, 41] (ascending) and [36, 26, 14, 6] (descending).**
    - **Bitonic Sequence: [11, 29, 19, 41, 36, 26, 14, 6]**

2) Bitonic Merge:

- **Merge the Bitonic Sequence [11, 29, 19, 41, 36, 26, 14, 6] in Ascending Order:**
    - **Compare 11 with 36, 29 with 26, 19 with 14, and 41 with 6.**
    - **Swap elements as needed:**
        - **After the first merge: [11, 26, 14, 6, 36, 29, 19, 41]**
- **Recursive Merging:**
    - **Continue recursively merging smaller parts until the entire array is fully sorted in ascending order.**
    - **Final Sorted Array: [6, 11, 14, 19, 26, 29, 36, 41]**

Source Code : https://github.com/Mohamed-Dawood/HPC-Project


File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
HPC Bitonic Sort
Debug
x64
Local Windows Debugger
Live Share
HPC Bitonic Sort.cpp
HPC Bitonic Sort
(Global Scope)
#include <iostream>
#include <vector>
// Function to compare and swap elements if needed
void compareAndSwap(std::vector<int>& arr, int i, int j, bool ascending) {
if (ascending == (arr[i] > arr[j])) {
std::swap(arr[i], arr[j]);
}
}
// Function to perform the bitonic merge
void bitonicMerge(std::vector<int>& arr, int low, int count, bool ascending) {
if (count > 1) {
int k = count / 2;
for (int i = low; i < low + k; i++) {
compareAndSwap(arr, i, i + k, ascending);
}
bitonicMerge(arr, low, k, ascending);
bitonicMerge(arr, low + k, k, ascending);
}
}
// Recursive function to generate a bitonic sequence and sort it
void bitonicSort(std::vector<int>& arr, int low, int count, bool ascending) {
if (count > 1) {
int k = count / 2;
// Sort the first half in ascending order
bitonicSort(arr, low, k, true);
// Sort the second half in descending order
bitonicSort(arr, low + k, k, false);
// Merge the bitonic sequence
bitonicMerge(arr, low, count, ascending);
}
}
int main() {
// Input array
std::vector<int> arr = { 11, 29, 19, 41, 14, 26, 36, 6 };
int n = arr.size();
Microsoft Visual Studio Debu
Sorted Array: [6, 11, 14, 19, 26, 29, 36, 41]
E:\C++ VS\HPC Bitonic Sort\x64\Debug\HPC Bitonic So
rt.exe (process 8756) exited with code 0.
Press any key to close this window . . .
Activate Windows
Go to Settings to activate Windows.
62 %
No issues found
Ln: 57 Ch: 1 SPC CRLF
Build succeeded
Add to Source Control
Select Repository
25°C
مشمس جزئيًا
Search
ENG
١١:٠٣ م
٢٠٢٤/١١/٠٩