Homework 4

Dzung Bui

Code of Task 1 – Homework 4

// Optimized implementation of Bubble sort

#include <stdio.h>

#include <cstdlib>

#include <iostream>

void swap(float \*xp, float \*yp)

{

float temp = \*xp;

\*xp = \*yp;

\*yp = temp;

}

// An optimized version of Bubble Sort

void bubbleSort\_variant(float arr[], int n)

{

int i, m, start, end, k, count;

// bool swapped;

start = 0;

end = 0;

count = 0;

bool still\_loop = true;

while (still\_loop)

{

++count;

// loop run from left to right

for (i = start; i < n-1-end; i++)

{

if (abs(arr[i]) > abs(arr[i+1]))

{ swap(&arr[i], &arr[i+1]); }

}

printf("Loop softing %d to the right/max side\n", count);

for (k=start; k < n-end; k++)

printf("%5.2f ", arr[k]);

printf("\n");

end++;

// for loop from right to left

for (m = n-1-end; m > start; m--)

{

if (abs(arr[m]) < abs(arr[m-1]))

{ swap(&arr[m], &arr[m-1]); }

}

start++;

if ((end+start) > n/2+2)

{

still\_loop = false;

// break;

}

printf("Loop sorting %d to the left/min side\n", count);

for (k=start-1; k < n-end; k++)

printf("%5.2f ", arr[k]);

printf("\n");

}

}

void sortCoupleNP(float arr[], int n)

{

bubbleSort\_variant(arr, n);

int i;

// loop run from left to right

for (i = 0; i < n-1; i++)

{

if (abs(arr[i]) == abs(arr[i+1]))

{

// printf("Couple negative and positive \n");

if (arr[i] > 0)

swap(&arr[i], &arr[i+1]);

}

}

}

/\* Function to print an array \*/

void printArray(float arr[], int size)

{

int i;

for (i=0; i < size; i++)

printf("%5.2f ", arr[i]);

printf("\n");

}

// Main program to test above functions

int main()

{

// input part

float arr[] = {4, 3, -2, 0, 2, 9, -1, 10, 0, 5, 23, -4,-23};

int n = sizeof(arr)/sizeof(arr[0]);

//print out the input data

printf("The input array: \n");

printArray(arr, n);

printf("\n");

// call the sorting function

sortCoupleNP(arr, n);

printf("\n");

printf("Sorted array: \n");

printArray(arr, n);

printf("\n");

return 0;

}

The output of the system:

