TOPIC NO.: 14(a)

TITLE: Bubble Sort (Normal)

Program Code:

#include <stdio.h>

void bubbleSort(int array[], int size)

{

int i, j, temp, count = 0;

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

{

for (j = 0; j < size - i - 1; j++)

{

count++;

if (array[j] > array[j + 1])

{

temp = array[j];

array[j] = array[j + 1];

array[j + 1] = temp;

}

}

}

printf("Step count: %d\n", count);

}

int main()

{

int size, count = 0;

printf("Enter the number of elements: ");

Output:

Enter the number of elements: 4

Enter 4 elements:

5

1

9

4

Original array: 5 1 9 4

Step count: 6

Sorted array: 1 4 5 9

DATE: 26.04.24

scanf("%d", &size);

int array[size];

printf("\nEnter %d elements:\n", size);

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

{

scanf("%d", &array[i]);

}

printf("\nOriginal array: ");

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

printf("%d ", array[i]);

}

printf("\n\n");

bubbleSort(array, size);

printf("\nSorted array: ");

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

printf("%d ", array[i]);

}

printf("\n\n");

return 0;

}

TOPIC NO.: 14(b)

TITLE: Bubble Sort (Optimized)

Program Code:

#include <stdio.h>

void bubbleSort(int array[], int size, int \*count)

{

int i, j, temp, swapped;

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

{

swapped = 0;

for (j = 0; j < size - i - 1; j++)

{

if (array[j] > array[j + 1])

{

temp = array[j];

array[j] = array[j + 1];

array[j + 1] = temp;

swapped = 1;

(\*count)++;

}

}

if (!swapped)

{

break;

}

}

}

int main()

{

int size, count = 0;

printf("Enter the number of elements: ");

scanf("%d", &size);

int array[size];

printf("Enter %d elements:\n", size);

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

{

scanf("%d", &array[i]);

}

DATE: 26.04.24

printf("Original array: ");

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

{

printf("%d ", array[i]);

}

printf("\n");

bubbleSort(array, size, &count);

printf("Sorted array: ");

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

{

printf("%d ", array[i]);

}

printf("\n");

printf("Total steps taken: %d\n", count);

return 0;

}

Output:

Enter the number of elements: 4

Enter 4 elements:

5

1

9

4

Original array: 5 1 9 4

Sorted array: 1 4 5 9

Total steps taken: 3