# c program for bubble sort

c program for bubble sort: c programming code for bubble sort to sort numbers or arrange them in ascending order. You can easily modify it to print numbers in descending order.

## Bubble sort algorithm in c

/\* Bubble sort code \*/

#include <stdio.h>

int main()

{

int array[100], n, c, d, swap;

printf("Enter number of elements\n");

scanf("%d", &n);

printf("Enter %d integers\n", n);

for (c = 0; c < n; c++)

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

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

{

for (d = 0 ; d < n - c - 1; d++)

{

if (array[d] > array[d+1]) /\* For decreasing order use < \*/

{

swap = array[d];

array[d] = array[d+1];

array[d+1] = swap;

}

}

}

printf("Sorted list in ascending order:\n");

for ( c = 0 ; c < n ; c++ )

printf("%d\n", array[c]);

return 0;

}

## Bubble sort in c language using function

#include <stdio.h>

void bubble\_sort(long [], long);

int main()

{

long array[100], n, c, d, swap;

printf("Enter number of elements\n");

scanf("%ld", &n);

printf("Enter %ld longegers\n", n);

for (c = 0; c < n; c++)

scanf("%ld", &array[c]);

bubble\_sort(array, n);

printf("Sorted list in ascending order:\n");

for ( c = 0 ; c < n ; c++ )

printf("%ld\n", array[c]);

return 0;

}

void bubble\_sort(long list[], long n)

{

long c, d, t;

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

{

for (d = 0 ; d < n - c - 1; d++)

{

if (list[d] > list[d+1])

{

/\* Swapping \*/

t = list[d];

list[d] = list[d+1];

list[d+1] = t;

}

}

}

}

**Compiler used:**

GCC

**Output of program:**

