void bubbleSort(vector<int>& arr) {

bool swapped = true;

int j = 0;

int n = arr.size();

//If at least one swap has been done, repeat

while (swapped) {

swapped = false;

j++;

for (int i = 0; i < n - j; i++) {

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

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

swapped = true;

}

}

}

}

Всяка итерация започва от елемент на позиция 0

като на финала й, най-големият елемент е „изплувал“

в края на масива чрез смяна на местата на елементите

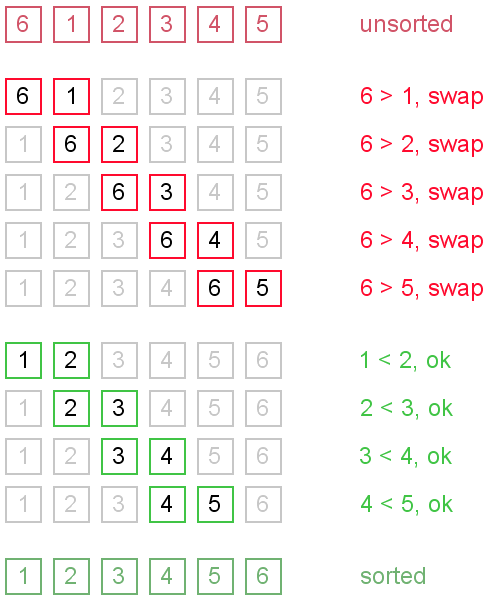
при наличие на съседска инверсия. Поради това,

следващата итерация ще приключи с 1 обхождане

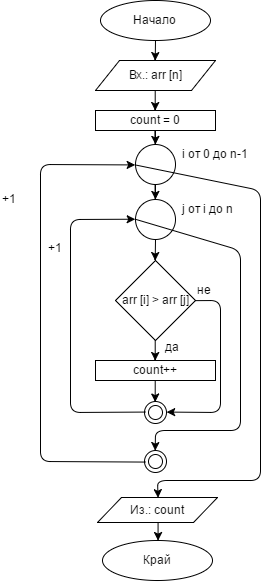
по-малко (последният елемент е вече на мястото си).

Процесът продължава докато се намира поне една

инверсия при обхождане на масива.



# Намиране броя на инверсиите в даден масив:

int getInvCount(vector<int> arr)

{

int count = 0;

int n = arr.size();

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

for (int j = i + 1; j < n; j++)

if (arr[i] > arr[j]) count++;

return count;

}