Оценка временной сложности функции пузырьковой сортировки.

void BubbleSort (int k,int x[max]) {

int i,j,buf;

for (i=k-1;i>0;i--)

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

if (x[j]>x[j+1]) {

buf=x[j];

x[j]=x[j+1];

x[j+1]=buf;

}

}

Оценим временную сложность функции пузырьковой сортировки в худшем случае, т.е. когда исходные данные отсортированы в обратном порядке. В этом случае внутренний цикл для каждого *i* выполнится *i-1* раз и произойдет обменов. Соответственно сложность алгоритма в худшем случае составит *O()* обменов.

Оценим временную сложность алгоритма пузырьковой сортировки в среднем случае, т.е. когда исходные данные имеют произвольный порядок. В этом случае условие во внутреннем цикле может выполниться *1,2,...,i-1* раз. Складывая, получим и, соответственно, условие во внутреннем цикле для каждого *i* выполнится в среднем раз и произойдет обменов. Соответственно сложность алгоритма в среднем случае составит *O()*.

В лучшем случае, т.е. когда исходные данные уже отсортированы. В этом случае будет выполнятся только один цикл без выполнения условия. Соответственно сложность алгоритма в лучшем случае составит *O(n)* обменов.