**Виды сортировок:**

Сортировка вставками:

void InsertSort(vector<T> &a) {

for (int i = 1; i < a.size(); i++) {

for (int j = i; j > 0 && a[j - 1] > a[j]; j--) {

swap(a[j - 1], a[j]);

count\_of\_swaps++;

count\_of\_compasions++;

}

}

}

Сортировка Шелла:

void ShellSort(vector<T> &a) {

for (int step = a.size() / 2; step > 0; step /= 2) {

for (int i = step; i < a.size(); i++) {

for (int j = i - step; j >= 0 && a[j] > a[j + step]; j -= step) {

T tmp;

tmp = a[j];

a[j] = a[j + step];

a[j + step] = tmp;

count\_of\_swaps++;

count\_of\_compasions++;

}

}

}

}

Сортировка Пузырьком:

void Bubble\_sort(vector<T> &a) {

for (int i = 0; i < a.size() - 1; i++) {

for (int j = 0; j < a.size() - i - 1; j++) {

count\_of\_compasions++;

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

int temp = a[j];

a[j] = a[j + 1];

a[j + 1] = temp;

count\_of\_swaps++;

}

}

}

}

Сортировка Чёт-Нечёт:

void OddEvenSort(vector<T> &a) {

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

for (int j = (i % 2) ? 0 : 1; j < a.size() - 1; j += 2) {

count\_of\_compasions++;

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

swap(a[j], a[j + 1]);

count\_of\_swaps++;

}

}

}

}

Шейкерная сортировка:

void ShakerSort(vector<T> &a) {

int left = 0;

int right = a.size() - 1;

T temp;

do {

for (int i = left; i < right; i++) {

count\_of\_compasions++;

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

temp = a[i];

a[i] = a[i + 1];

a[i + 1] = temp;

count\_of\_swaps++;

}

}

right--;

for (int i = right; left < i; i--) {

count\_of\_compasions++;

if (a[i] < a[i - 1]) {

temp = a[i];

a[i] = a[i - 1];

a[i - 1] = temp;

count\_of\_swaps++;

}

}

left++;

} while (left < right);

}

Сортировка Расчёской:

void CombSort(vector<T> &a) {

const double factor = 1.247;

double gapFactor = a.size() / 1.247;

while (gapFactor > 1) {

const double gap = round(gapFactor);

for (int i = 0, j = gap; j < a.size(); i++, j++) {

count\_of\_compasions++;

if (a[i] >= a[j]) {

swap(a[i], a[j]);

count\_of\_swaps++;

}

}

gapFactor = gapFactor / factor;

}

}

Сортировка Выбором:

void ChooseSort(vector<T> &a) {

T tmp;

int k;

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

k = i;

tmp = a[i];

for (int j = i + 1; j < a.size(); j++) {

count\_of\_compasions++;

if (a[j] < tmp) {

k = j;

tmp = a[j];

count\_of\_swaps++;

}

}

a[k] = a[i];

a[i] = tmp;

}

}

Сортировка Двойным Выбором:

void DoubleChooseSort(vector<T> &arr) {

for (int i = 0, j = arr.size() - 1; i < j; i++, j--) {

T min = arr[i], max = arr[i];

int minIndex = i, maxIndex = i;

for (int k = i; k <= j; k++) {

if (arr[k] > max) {

max = arr[k];

maxIndex = k;

count\_of\_compasions++;

} else if (arr[k] < min) {

min = arr[k];

minIndex = k;

count\_of\_compasions++;

}

}

count\_of\_swaps++;

swap(arr[i], arr[minIndex]);

if (arr[minIndex] == max) {

swap(arr[j], arr[minIndex]);

count\_of\_swaps++;

count\_of\_compasions++;

} else {

swap(arr[j], arr[maxIndex]);

count\_of\_swaps++;

count\_of\_compasions++;

}

}

}