#include <iostream>

#include <vector>

using namespace std;

// Fungsi untuk menampilkan array

void printArray(vector<int>& arr) {

for (int num : arr) {

cout << num << " ";

}

cout << endl;

}

// Insertion Sort

void insertionSort(vector<int>& arr, int& comparisons, int& shifts) {

int n = arr.size();

comparisons = 0;

shifts = 0;

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

int key = arr[i];

int j = i - 1;

while (j >= 0 && arr[j] > key) {

arr[j + 1] = arr[j];

j--;

shifts++;

comparisons++;

}

arr[j + 1] = key;

shifts++;

}

}

// Binary Insertion Sort

void binaryInsertionSort(vector<int>& arr, int& comparisons, int& shifts) {

int n = arr.size();

comparisons = 0;

shifts = 0;

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

int key = arr[i];

int left = 0, right = i;

while (left < right) {

int mid = left + (right - left) / 2;

if (key < arr[mid]) {

right = mid;

}

else {

left = mid + 1;

}

comparisons++;

}

for (int j = i; j > left; j--) {

arr[j] = arr[j - 1];

shifts++;

}

arr[left] = key;

shifts++;

}

}

// Selection Sort

void selectionSort(vector<int>& arr, int& comparisons, int& shifts) {

int n = arr.size();

comparisons = 0;

shifts = 0;

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

int minIndex = i;

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

if (arr[j] < arr[minIndex]) {

minIndex = j;

}

comparisons++;

}

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

shifts += 3; // Setiap swap membutuhkan 3 pergeseran

}

}

int main() {

vector<int> arr = { 64, 25, 12, 22, 11 };

int comparisons, shifts;

cout << "Insertion Sort:" << endl;

insertionSort(arr, comparisons, shifts);

printArray(arr);

cout << "Perbandingan: " << comparisons << ", Pergeseran: " << shifts << endl;

cout << endl;

arr = { 64, 25, 12, 22, 11 };

cout << "Binary Insertion Sort:" << endl;

binaryInsertionSort(arr, comparisons, shifts);

printArray(arr);

cout << "Perbandingan: " << comparisons << ", Pergeseran: " << shifts << endl;

cout << endl;

arr = { 64, 25, 12, 22, 11 };

cout << "Selection Sort:" << endl;

selectionSort(arr, comparisons, shifts);

printArray(arr);

cout << "Perbandingan: " << comparisons << ", Pergeseran: " << shifts << endl;

return 0;

}