**Kode Percobaan & Latihan dalam Bentuk WORD**

Praktikum Algoritma dan Struktur Data

****

**Oleh:**

Samudero Dirgantoro / 5223600016

**Program Studi D4 Teknologi Game**

**Departemen Teknologi Multimedia Kreatif**

**Politeknik Elektronika Negeri Surabaya**

**2023/2024**

1. Percobaan:

//Samudero Dirgantoro 5223600016

#include <iostream>

#include <cstdlib>

#include <ctime>

#define MAX 10

using namespace std;

int Data[MAX];

// Pertukaran bilangan

void Tukar(int\* a, int\* b) {

int temp = \*a;

\*a = \*b;

\*b = temp;

}

// Pengurutan metode Bubble

void Bubble() {

int i, j;

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

for (j = 0; j < MAX - i - 1; j++) {

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

Tukar(&Data[j], &Data[j + 1]);

}

}

}

}

// Pengurutan metode Shell

void Shell() {

int i, j, gap;

for (gap = MAX / 2; gap > 0; gap /= 2) {

for (i = gap; i < MAX; i++) {

for (j = i - gap; j >= 0 && Data[j] > Data[j + gap]; j -= gap) {

Tukar(&Data[j], &Data[j + gap]);

}

}

}

}

void TampilkanData(const char\* label) {

cout << label << "\n";

for (int i = 0; i < MAX; i++) {

cout << "Data ke " << i << " : " << Data[i] << "\n";

}

}

int main() {

int pilihan;

// Generate bilangan acak

srand(time(0));

cout << "Membangkitkan bilangan acak\n";

for (int i = 0; i < MAX; i++) {

Data[i] = rand() % 1000 + 1;

}

cout << "Pilih metode pengurutan:\n 1. Pengurutan Bubble\n 2. Pengurutan Shell\n Masukkan pilihan Anda: ";

cin >> pilihan;

// Menu

switch (pilihan) {

case 1:

Bubble();

TampilkanData("Bubble:");

break;

case 2:

Shell();

TampilkanData("Shell:");

break;

default:

std::cout << "Invalid.\n";

break;

}

}

1. Latihan:

#include <iostream>

#include <string>

using namespace std;

struct Pegawai {

int NIP;

string Nama;

string Alamat;

char Golongan;

};

// Fungsi untuk menampilkan data pegawai

void tampilkanPegawai(Pegawai arr[], int n) {

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

cout << "NIP: " << arr[i].NIP << ", Nama: " << arr[i].Nama << ", Alamat: " << arr[i].Alamat << ", Golongan: " << arr[i].Golongan << "\n";

}

cout << "\n";

}

// Fungsi Bubble berdasarkan NIP

void bubbleNIP(Pegawai arr[], int n, bool urutanNaik) {

int i, j;

Pegawai temp;

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

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

if (urutanNaik) {

if (arr[j].NIP > arr[j + 1].NIP) {

temp = arr[j];

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

arr[j + 1] = temp;

}

}

else {

if (arr[j].NIP < arr[j + 1].NIP) {

temp = arr[j];

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

arr[j + 1] = temp;

}

}

}

}

tampilkanPegawai(arr, n);

}

// Fungsi Bubble berdasarkan Nama

void bubbleNAMA(Pegawai arr[], int n, bool urutanNaik) {

int i, j;

Pegawai temp;

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

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

if (urutanNaik) {

if (arr[j].Nama > arr[j + 1].Nama) {

temp = arr[j];

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

arr[j + 1] = temp;

}

}

else {

if (arr[j].Nama < arr[j + 1].Nama) {

temp = arr[j];

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

arr[j + 1] = temp;

}

}

}

}

tampilkanPegawai(arr, n);

}

// Fungsi Shell berdasarkan NIP

void shellNIP(Pegawai arr[], int n, bool urutanNaik) {

int i, j, gap;

Pegawai temp;

for (gap = n / 2; gap > 0; gap /= 2) {

for (i = gap; i < n; i++) {

temp = arr[i];

if (urutanNaik) {

for (j = i; j >= gap && arr[j - gap].NIP > temp.NIP; j -= gap) {

arr[j] = arr[j - gap];

}

}

else {

for (j = i; j >= gap && arr[j - gap].NIP < temp.NIP; j -= gap) {

arr[j] = arr[j - gap];

}

}

arr[j] = temp;

}

}

tampilkanPegawai(arr, n);

}

// Fungsi Shell berdasarkan Nama

void shellNAMA(Pegawai arr[], int n, bool urutanNaik) {

int i, j, gap;

Pegawai temp;

for (gap = n / 2; gap > 0; gap /= 2) {

for (i = gap; i < n; i++) {

temp = arr[i];

if (urutanNaik) {

for (j = i; j >= gap && arr[j - gap].Nama > temp.Nama; j -= gap) {

arr[j] = arr[j - gap];

}

}

else {

for (j = i; j >= gap && arr[j - gap].Nama < temp.Nama; j -= gap) {

arr[j] = arr[j - gap];

}

}

arr[j] = temp;

}

}

tampilkanPegawai(arr, n);

}

int main() {

// Contoh data pegawai

Pegawai pegawai[] = {

{101, "Dimas", "Jakarta", 'A'},

{103, "Anwar", "Bandung", 'B'},

{102, "Budi", "Surabaya", 'C'},

{104, "Jojo", "Yogyakarta", 'A'}

};

int n = sizeof(pegawai) / sizeof(pegawai[0]);

// Pilihan metode pengurutan

int metode;

cout << "Pilih metode pengurutan:\n";

cout << "1. Bubble Sort\n";

cout << "2. Shell Sort\nInput: ";

cin >> metode;

// Pilihan urutan

int urutan;

cout << "Pilih urutan:\n";

cout << "1. Naik\n";

cout << "2. Turun\nInput: ";

cin >> urutan;

// Pilihan pengurutan berdasarkan

int berdasarkan;

cout << "Pilih pengurutan berdasarkan:\n";

cout << "1. NIP\n";

cout << "2. Nama\nInput: ";

cin >> berdasarkan;

// Mengurutkan data berdasarkan pilihan

switch (berdasarkan) {

case 1:

switch (metode) {

case 1:

bubbleNIP(pegawai, n, (urutan == 1));

break;

case 2:

shellNIP(pegawai, n, (urutan == 1));

break;

default:

cout << "Pilihan tidak valid.";

}

break;

case 2:

switch (metode) {

case 1:

bubbleNAMA(pegawai, n, (urutan == 1));

break;

case 2:

shellNAMA(pegawai, n, (urutan == 1));

break;

default:

cout << "Pilihan tidak valid.";

}

break;

default:

cout << "Pilihan tidak valid.";

}

return 0;

}