Tugas Praktikum 11

1. Usaha untuk mengurutkan kumpulan-kumpulan data dalam suatu array disebut:
2. Searching
3. Sorting
4. Divide
5. Conquer
6. Berikut ini adalah metode yang digunakan pada teknik sorting, kecuali:
   1. Bubble
   2. Fibonacci
   3. Heap
   4. Radix
7. Data 4 0 8 2, diurutkan secara ascending (dari kecil ke besar) dengan metode bubble sort. Hasil urutan data pass satu (tahap satu) adalah:
   1. 0 4 2 8
   2. 0 8 2 4a
   3. 0 4 8 2
   4. 0 2 4 8
8. Pada data 0 6 3 2 4, akan dilakukan maximum selection sort secara ascending, maka pada langkah pertama, urutan data yang terjadi adalah:
   1. 0 2 6 3 4
   2. 0 4 3 2 6
   3. 0 2 3 4 6
   4. 0 3 6 2 4
9. Pengurutan adalah proses untuk:
   1. mencari elemen sebuah list
   2. mengecek harga elemen tertentu
   3. pengaturan kembali elemen kedalam urutan tertentu
   4. menyederhanakan persoalan dengan cara memecah ke persoalan yang lebih kecil
10. Misal terdapat record mahasiswa sebagai berikut : nim mahasiswa (tipe data integer), nama mahasiswa (tipe data string, panjang max 15 char), tahun lahir (tipe integer),umur (tipe real). Buat algoritma dengan menggunakan procedure berparameter untuk menampilkan kembali data mahasiswa yang terurut berdasarkan tahun lahir. Asumsikan terdapat 5 mahasiswa.

#include<iostream>

#include<string>

using namespace std;

// membuat struktur data untuk mahasiswa yang di deklarasikan berdasarkan tipe data dan variabelnya. yaitu nim,lahir,umur,nama dengan banyak index sebanyak 5

typedef struct data{

int nim[5], lahir[5], umur[5];

string nama[5];

}data;

// fungsi untuk menampilkan data mahasiswa yang data nya diambil dari inputan keyboard yang ada di int main

void data\_mahasiswa(int nim[], string nama[], int lahir[], int umur[], int n)

{

for(int i=0; i<n; i++)//looping berdasarkan banyak data dan isi array yang ada dari inputan keyboard di int main

{

cout << "Nama Mahasiswa ke-" << i+1 << " : " << nama[i] << endl;

cout << "NIM " << nama[i] << " : " << nim[i] << endl;

cout << "Tahun Lahir " << nama[i] << " : " << lahir[i] << endl;

cout << "Umur " << nama << " : " << umur[i] << endl;

}

}

// fungsi untuk mengurutkan data mahasiswa berdasarkan tahun lahir

void sort\_data(int nim[], string nama[], int lahir[], int umur[], int n)

{

for(int i=0; i<n; i++)//looping untuk meloping array dari index nol sampai index ke n

{

for(int j=0; j<n-i-1; j++)//looping untuk meloping array dari index perpindahan sampai index ke n

{

if(lahir[j] > lahir[j+1])

{

// menukar data jika tahun lahir pada indeks ke-j lebih besar dari tahun lahir pada indeks ke-j+1

int temp = lahir[j];

lahir[j] = lahir[j+1];

lahir[j+1] = temp;

temp = nim[j];

nim[j] = nim[j+1];

nim[j+1] = temp;

temp = umur[j];

umur[j] = umur[j+1];

umur[j+1] = temp;

string temp2 = nama[j];

nama[j] = nama[j+1];

nama[j+1] = temp2;

}

}

}

}

int main()

{//memanggil tipe data yang sudah distrukturkan oleh typedef struc dan di definiskan menjadi asum

data asum;

int n=5;

// menginput data mahasiswa sebanyak looping n

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

{

cout << "Masukan Nama Mahasiswa ke-" << i+1 << " : ";

getline(cin,asum.nama[i]);

asum.nama[i] = asum.nama[i].substr(0, 15); // membatasi nama hanya dapat diinput sebanyak 15 karakter

cout << "Masukan NIM Mahasiswa ke-" << i+1 << " : ";

cin >> asum.nim[i];

cout << "Masukan Tahun Lahir Mahasiswa ke-" << i+1 << " : ";

cin >> asum.lahir[i];

cout << "Masukan Umur Mahasiswa ke-" << i+1 << " : ";

cin >> asum.umur[i];

cin.ignore(); // mengabaikan newline character dari cin, sehingga dapat mengisi inputan keyboard berikutnya sampai looping selesai

}

// mengurutkan data mahasiswa dan menampilkan data mahasiswa yang di lemparkan datanya ke fungsi sort\_data dan data\_mahasiswa

sort\_data(asum.nim, asum.nama, asum.lahir, asum.umur, n);

data\_mahasiswa(asum.nim, asum.nama, asum.lahir, asum.umur, n);

return 0;

}

1. PT. MURAH HATI memberi komisi salesmannya berdasarkan ketentuan sebagai berikut :
   * + Bila salesman berhasil menjual barang ***seharga*** Rp 500.000,- maka dia akan mendapat komisi sebesar 10 %.
     + Bila lebih dari Rp 500.000,-, untuk Rp 500.000,- pertama komisinya 10 %, sedangkan sisanya mendapat 15 %.

Bila perusahaan tersebut memiliki 5 orang salesman, rancanglah algoritma untuk menentukan komisi yang diterima oleh ***setiap*** salesmannya, serta ***total komisi*** yang telah dibayarkan oleh PT. MURAH HATI kepada ke 5 salesman tadi serta total penjualan yang berhasil dilakukan para salesman.

Adapun spesifikasi dari algoritma adalah:

1. Algoritma harus menggunakan *procedure* atau *function* **berparameter**.
2. Output yang diinginkan adalah memunculkan data karyawan (nama karyawan, besar penjualan, serta komisi) yang **terurut secara *descending* berdasarkan besar penjualan**.

#include<iostream>

#include<string>

using namespace std;

//merupakan kelas yang mendeklarasikan tipe data yang akan menyimpan data

class Data {

public:

string nama[5];

int pemasukan[5];

};

//ini merupakan bubble sort yang akan mendapatkan data dari inputan keyboard yang ada di int main.

//nama[]=inputan asum.nama

//pendapatan[]=inputan asum.pemasukan

void sort\_data(string nama[], int pendapatan[], int n) {

//melakukan pengulangan

for(int i = 0; i < n; i++) {//pengulangan pertama : berfungsi untuk mengulang dari data awal sampai akhir

for(int j = 0; j < n - i - 1; j++) {//pengulangan kedua : berfungsi untk mengulang data yang kedua setelah terjadi perpindahan

if(pendapatan[j] < pendapatan[j + 1]) {

//dibuatkan 1 tempat untuk menyimpan data terbesar di paling kiri berdasarkan array. contoh. 123 -> 2 lebih besar dari 1 maka 2 dibuatkan di tempat terbaru dan dilanjutkan angka 1. begitu seterusnya

int temp = pendapatan[j];

pendapatan[j ] = pendapatan[j+1];

pendapatan[j+1] = temp;

string temp2 = nama[j ];

nama[j ] = nama[j+1];

nama[j+1] = temp2;

}

}

}

}

//digunakan untuk oprasi perhitungan komisi, dengan data pemasukan didapatkan dari inputan asum.pemasukan

double komisi(int pemasukan) {

//dideklarasikan batas sebagai batas perkalian, dan j bertugas untuk mengembalikan data ke intmain dan operasi perkalian berdasarkan persentase dan batas

int batas = 500000;

double j = 0;

if(pemasukan == batas) {//jika batas==pemasukan atau pendapatan nya mencapai 500000 maka akan dapat komisi sebesar batas\*0.1

j = batas \* 0.1;

} else if(pemasukan > batas) {//jika melebihi batas maka, kelebihan dari batas itu akan dikalikan 15% ditambah dengan pencapaian standarnya

j = (batas \* 0.1) + ((pemasukan - batas) \* 0.15);

}

return j;//mengembalikan nilai j

}

//outputan data yang telah diurutkan oleh bubble sort dan memasukkannya ke fungsi data\_karyawan

void data\_karyawan(string nama[], int pemasukan[], int n) {cout<<"======PT. MURAH HATI ======"<<endl;

for(int i = 0; i < n; i++) {//melakukan pengulangan untuk mengoutputkan karyawan dan pendapatan berdasarkan banyak array yang diinputkan di intmain

double k = komisi(pemasukan[i]);//menggunakan fungsi komisi untuk dapat di outputkan dan di deklarasikan menjadi tipe data double yang bervariabel k

cout << "Nama: " << nama[i] << "\nPendapatan: " << pemasukan[i] << "\nKomisi: " << k << endl;

}

}

int main() {

Data asum;//memanggil tipe data dan variabel yang akan digunakan

int n = 5;//menetapkan banyak data yang akan diloop sebanyak n

for(int i = 0; i < n; i++) {//looping inputan keyboard yang akan disimpan dalam array berdasarkan variabel yang telah dideklarasikan di class

cout << "Masukkan Nama Karyawan ke-" << i + 1 << " : ";

getline(cin, asum.nama[i]);//untuk menyalin semua inputan keyboard dengan tipe data string yang akan disimpan di array berdasarkan variabelnya

cout << "Masukkan Pendapatan ke-" << i + 1 << " : ";

cin >> asum.pemasukan[i];//untuk menyalin semua inputan keyboard dengan tipe data int

cin.ignore(); // mengabaikan newline character dari cin sehingga ketika di loop, tidak membuat data inputan selanjutnya eror, karena mengisi data dari inputan sebelumnya

}

//memasukan data inputan ke fungsi sor\_data untuk mensorting data komisi dari besar ke kecil berdasarkan data penjualan per karyawan

sort\_data(asum.nama, asum.pemasukan, n);

//memasukan data inputan yang telah di sorting, untuk di outputkan menggunakan fungsi data\_karyawan

data\_karyawan(asum.nama, asum.pemasukan, n);

return 0;

}