#include"listOfPegawai.h"

void createListPgw(ListOfPgw \*LP){

(\*LP).first=NULL;

}

//create list kosong

address alokasi(int nipX, char\* namaX, int transportX, int makanX, int gajiX){

address p=(address)malloc(sizeof(ElmtList));

p->nip=nipX;

p->nama=namaX;

p->transportasi=transportX;

p->makan=makanX;

p->gaji=gajiX;

p->totalBiaya=NULL;

p->next=NULL;

}

//ketika mengisikan nilai ke field totalBiaya, isilah dengan nilai 0 (nol), karena akan diupdate oleh procedure processTotalPendapatan

void dealokasi(address p){

free(p);

}

void insertFirstPgw(ListOfPgw \*LP, address p){

p->next=(\*LP).first;

(\*LP).first=p;

}

void insertLastPgw(ListOfPgw \*LP, address p){

address last=(\*LP).first;

if(last!=NULL){

while((last)->next!=NULL){

last=last->next;

}

last->next=p;

}else{

insertFirst(\*LP,p);

}

}

void deleteFirstPgw(ListOfPgw \*LP){

address p=(\*LP).first;

(\*LP).first=p->next;

free(p);

}

void deleteLastPgw(ListOfPgw \*LP){

address p=(\*LP).first;

while(p->next!=NULL)

p=p->next;

free(p);

}

void prosesTotalBiaya(ListOfPgw \*LP); //mengisi field totalBiaya sebagai hasil penjumlahan dari field transportasi, makan, dan gaji

void cetakListPgw(ListOfPgw LP){ // mencetak nip, nama, biaya transportasi, makan, gaji, dan totalBiaya dari semua pegawai

address p=LP.first;

while(p!=NULL){

printf("\nNIP: %i\n"

"Nama: %s\n"

"Biaya Transportasi: %d\n"

"Makan: %d\n"

"Gaji: %d\n"

"Nilai: %i\n"

"Total Biaya: %d\n"

,p->nip,p->nama,p->transportasi,p->makan,p->gaji,p->totalBiaya);

p=p->next;

}

}

address searchPeg(ListOfPgw LP, int nipX){

/\* melakukan pencarian berdasarkan NIP, bila ketemu maka return-lah address-nya, bila tidak ketemu maka return NULL

NIP bersifat unik (tidak ada yang sama)

\*/

address p=LP.first;

while(p->next!=NULL||p->nip!=nipX)

p=p->next;

if(p->nip==nipX)

return p;

else

return NULL;

}

int nbElmt(ListOfPgw LP){ //menghitung banyaknya elemen di dalam list

address p=LP.first;

int n=1;

while(p->next!=NULL)

n++;

return n;

}

int maxGaji(ListOfPgw LP){ //mencari nilai tertinggi dari field gaji

int max;

address p=LP.first;

max=p->gaji;

while(p->next!=NULL){

if(max<p->gaji)

max=p->gaji;

p=p->next;

}

return max;

}

int minGaji(ListOfPgw LP){ //mencari nilai terendah dari field gaji

int min;

address p=LP.first;

min=p->gaji;

while(p->next!=NULL){

if(min>p->gaji)

min=p->gaji;

p=p->next;

}

return min;

}

float avgGaji(ListOfPgw LP){ //mencari nilai rata-rata gaji seluruh pegawai

return sumTotalBiaya(LP)/nbElmt(LP);

}

int sumTotalBiaya(ListOfPgw LP){ //menjumlahkan totalBiaya seluruh pegawai

int jumlah=0;

address p=LP.first;

while(p->next!=NULL){

jumlah=jumlah+p->gaji;

p=p->next;

}

return jumlah;

}

**"Saya mengerjakan tugas ini dengan jujur dan tidak melakukan pelanggaran aturan dari IFLAB.**

**Jika saya terbukti melanggar, maka saya bersedia menerima sanksi yang berlaku."**