**HEADER**

//1302154158\_SALMA

#include <stdio.h>

#include <malloc.h>

#include <string.h>

#include "boolean.h"

#ifndef LISTREKURSIF\_H\_INCLUDED

#define LISTREKURSIF\_H\_INCLUDED

typedef struct tElmtlist \*address;

typedef struct tElmtlist{

char\* namaPenyewa; //diasumsikan unik (tidak ada yang sama)

char\* hariMasuk; //hari check in

char\* hariKeluar; //hari check out

int hargaTotal;

address next;

}Elmtlist;

typedef address ListOfHotel;

void makeListBill(ListOfHotel \*LH, char\* namaX, char\* hariMasukX, char\* harikeluarX, int hargaTotalX);

//membuat list dengan isi masukan pertama

void insertLastBill(ListOfHotel \*LH, char\* namaX, char\* hariMasukX, char\* harikeluarX, int hargaTotalX);

void cetakList(ListOfHotel LH); //mencetak nama penyewa, hari masuk, hari keluar, harga total dari semua penyewa

boolean cariOrang(ListOfHotel LH, char\* nama);

//pencarian berdasarkan nama, bila ketemu maka return address, bila tidak ketemu return NULL

int nbElmt(ListOfHotel LH); //menghitung banyaknya elemen di dalam list

int maxBill(ListOfHotel LH); //mencari nilai tertinggi dari harga total

int minBill(ListOfHotel LH); //mencari nilai terendah dari harga total

float avgBill(ListOfHotel LH); //mencari nilai rata-rata dari harga total seluruh penyewa

int sumBill(ListOfHotel LH); //menjumlahkan total harga total seluruh penyewa

#endif // LISTREKURSIF\_H\_INCLUDED

**DRIVER**

//1302154158\_SALMA

#include <stdio.h>

#include "listrekursif.c"

int main(){

ListOfHotel lh;

makeListBill(&lh, "Szczesny", "Senin", "Rabu", 350000);

insertLastBill(&lh, "Blaszczykowski", "Senin", "Rabu", 350000);

insertLastBill(&lh, "Matuszczyk", "Selasa", "Sabtu", 675000);

insertLastBill(&lh, "Jedrzejczyk", "Minggu", "Senin", 200000);

insertLastBill(&lh, "Chygrynsky", "Rabu", "Jumat", 400000);

cetakList(lh);

printf("\n[0=No,1=Yes] search(lk,Blaszczkowski) = %i\n\n", cariOrang(lh, "Blaszczkowski"));

printf("nbElmt = %i\n", nbElmt(lh));

printf("maxBill = %i\n", maxBill(lh));

printf("minBill = %i\n", minBill(lh));

printf("avgBill = %f\n", avgBill(lh));

printf("sumBill = %i\n", sumBill(lh));

return 0;

}

**BODY**

//1302154158\_SALMA

#include "listrekursif.h"

void makeListBill(ListOfHotel \*LH, char\* namaX, char\* hariMasukX, char\* harikeluarX, int hargaTotalX){//membuat list dengan isi masukan pertama

address a = (address)malloc(sizeof(Elmtlist));

if(a!=NULL){

a->namaPenyewa=namaX;

a->hariMasuk=hariMasukX;

a->hariKeluar=harikeluarX;

a->hargaTotal=hargaTotalX;

a->next=NULL;

}

(\*LH)=a;

}

void insertLastBill(ListOfHotel \*LH, char\* namaX, char\* hariMasukX, char\* harikeluarX, int hargaTotalX){

if((\*LH)->next!=NULL){

insertLastBill(&((\*LH)->next),namaX,hariMasukX,harikeluarX,hargaTotalX);

}else{

ListOfHotel b=(\*LH);

makeListBill(&b,namaX,hariMasukX,harikeluarX,hargaTotalX);

(\*LH)->next=b;

}

}

void cetakList(ListOfHotel LH){ //mencetak nama penyewa, hari masuk, hari keluar, harga total dari semua penyewa

if(LH!=NULL){

printf(" Nama Penyewa : %s\n Hari Masuk : %s\n Hari Keluar : %s\n Harga Total : %d \n",LH->namaPenyewa,LH->hariMasuk,LH->hariKeluar,LH->hargaTotal);

cetakList(LH->next);

}

}

boolean cariOrang(ListOfHotel LH, char\* nama){//pencarian berdasarkan nama, bila ketemu maka return address, bila tidak ketemu return NULL

if(LH==NULL){

return false;

}else{

if(!strcmp(LH->namaPenyewa,nama)){

return true;

}else{

return cariOrang(LH->next,nama);

}

}

}

int nbElmt(ListOfHotel LH){ //menghitung banyaknya elemen di dalam list

if(LH==NULL){

return 0;

}else{

return(1+nbElmt(LH->next));

}

}

int maxBill(ListOfHotel LH){ //mencari nilai tertinggi dari harga total

if(LH==NULL){

return 0;

}else{

if((LH)->hargaTotal>maxBill(LH->next)){

return LH->hargaTotal;

}else{

return maxBill(LH->next);

}

}

}

int minBill(ListOfHotel LH){ //mencari nilai terendah dari harga total

if(LH==NULL){

return 0;

}else{

if(LH->hargaTotal<maxBill(LH->next)){

return (LH)->hargaTotal;

}else{

return maxBill(LH->next);

}

}

}

float avgBill(ListOfHotel LH){ //mencari nilai rata-rata dari harga total seluruh penyewa

return(sumBill(LH)/nbElmt(LH));

}

int sumBill(ListOfHotel LH){ //menjumlahkan total harga total seluruh penyewa

if(LH==NULL){

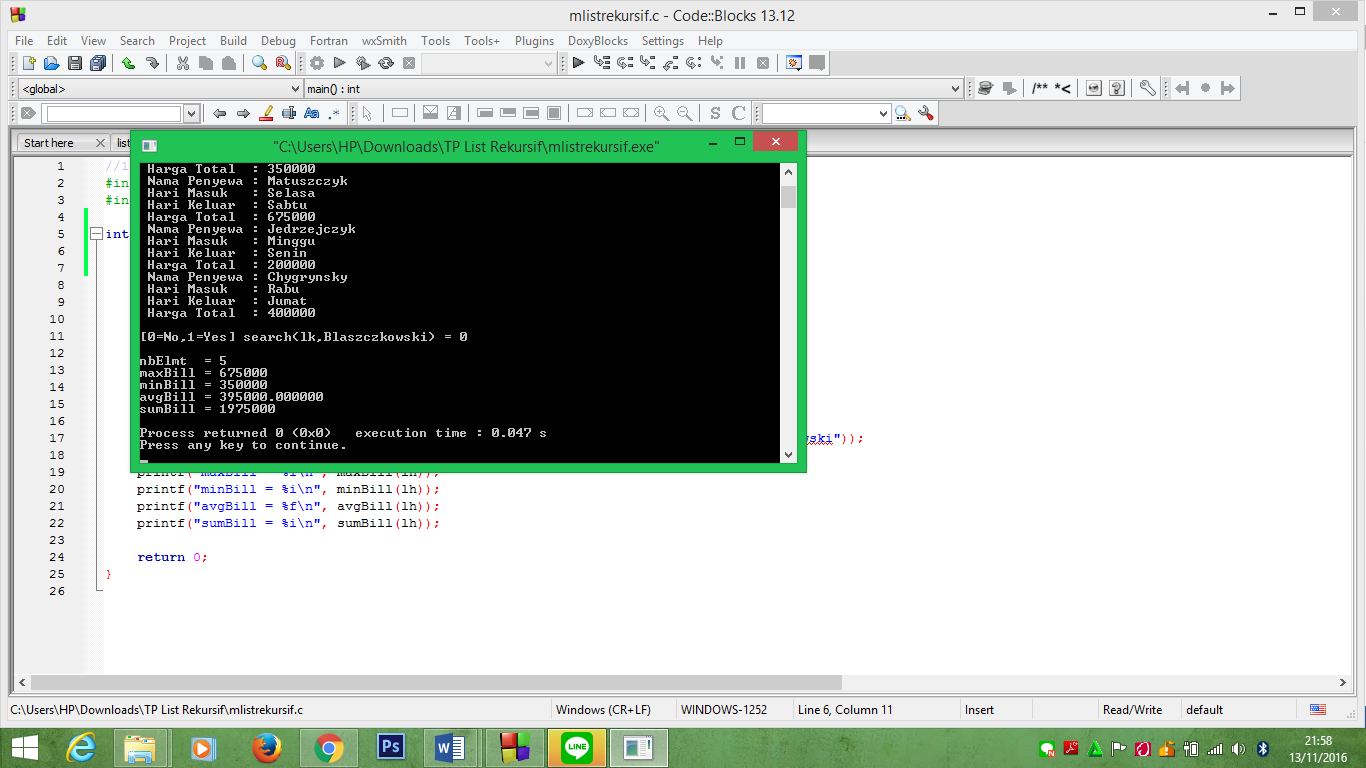
return 0;

}else{

return(LH->hargaTotal+sumBill(LH->next));

}

}



+==================================================================+

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

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

+==================================================================+