Pseudocode

A. Projek

Algoritma Aplikasi\_Booking\_Hotel\_Online

Algoritma program untuk memberikan kemudahan dalam memesan hotel secara online di Hotel 9 dengan memberikan pilihan kamar beserta harga total dalam transaksi. Array nama, lamainap, kamar, bayar berguna untuk menyimpan bobot input, variabel struct sewa, totalharga, hargaawal, lamainap bertipe integer, variabel struct looping, nama, pilihan, pilihan2, kamar bertipe char, variabel struct jeniskamar, bayar bertipe real.

[Deklarasi Struct]

Struct jeniskamar{

const double a = 250000;

const double b = 350000;

const double c = 500000;

const double d = 750000;

const double e = 900000;

};

Struct sewa{

jeniskamar a;

jeniskamar b;

jeniskamar c;

jeniskamar d;

jeniskamar e;

int harisewa;

};

Struct looping{

char pilihan2;

};

[Deklarasi Array]

CHAR nama [10][50]

INTEGER lamainap [10]

CHAR kamar [10]

DOUBLE bayar [10]

[Deklarasi Fungsi]

FLOAT float room\_a (float harga, int harisewa)

FLOAT float room\_b (float harga, int harisewa)

FLOAT float room\_c (float harga, int harisewa)

FLOAT float room\_d (float harga, int harisewa)

FLOAT float room\_e (float harga, int harisewa)

VOID booking (int i)

[Deklarasi Variabel Struct dalam VOID]

STRUCT :

jeniskamar jk

sewa sk

1. [Mulai]

[Masukkan nama penyewa]

Read (nama[i])

1. [Masukkan lama hari menginap]

Read (sk.harisewa)

lamainap[i] = sk.harisewa

1. [Masukkan jenis pilihan ruangan yang diinginkan]

Read (pilihan)

Kamar[i] = pilihan

1. [penulisan switch case]

switch(pilihan){

case 'a' : {

hargaawal = jk.a \* sk.harisewa;

totalharga = room\_a (jk.a, sk.harisewa);

write ( ‘ Harga sebelum dikenakan pajak sebesar 10%: Rp. ‘ hargaawal)

write( ‘ total harga yang harus dibayarkan adalah : Rp. ‘ totalharga)

bayar[i] = totalharga;

break;

}

case 'b' : {

hargaawal = jk.b \* sk.harisewa;

totalharga = room\_b (jk.b, sk.harisewa);

write ( ‘ Harga sebelum dikenakan pajak sebesar 10%: Rp. ‘ hargaawal)

write( ‘ total harga yang harus dibayarkan adalah : Rp. ‘ totalharga)

bayar[i] = totalharga;

break;

}

case 'c' : {

hargaawal = jk.c \* sk.harisewa;

totalharga = room\_c (jk.c, sk.harisewa);

write ( ‘ Harga sebelum dikenakan pajak sebesar 10%: Rp. ‘ hargaawal)

write( ‘ total harga yang harus dibayarkan adalah : Rp. ‘ totalharga)

bayar[i] = totalharga;

break;

}

case 'd' : {

hargaawal = jk.d \* sk.harisewa;

totalharga = room\_d (jk.d, sk.harisewa);

write ( ‘ Harga sebelum dikenakan pajak sebesar 10%: Rp. ‘ hargaawal)

write( ‘ total harga yang harus dibayarkan adalah : Rp. ‘ totalharga)

bayar[i] = totalharga;

break;

}

case 'e' : {

hargaawal = jk.e \* sk.harisewa;

totalharga = room\_e (jk.e, sk.harisewa);

write ( ‘ Harga sebelum dikenakan pajak sebesar 10%: Rp. ‘ hargaawal)

write( ‘ total harga yang harus dibayarkan adalah : Rp. ‘ totalharga)

bayar[i] = totalharga;

break;

}

default : {

write ( ‘pilihan tidak valid’ )

bayar[i] = 0;

break;

}

1. [Algoritma utama]

[Deklarasi variabel dalam algoritma utama]

STRUCT looping dlc

INTEGER I = 0

VOID booking (i)

[Masukkan pilihan untuk memilih apakah ingin memesan lagi atau tidak]

Read (dlc.pilihan2)

1. [Struktur pengulangan untuk menawarkan untuk ingin memesan lagi atau tidak]

while(dlc.pilihan2 == 'Y'||dlc.pilihan2 == 'y'){

i++;

booking (i);

write ( ‘Apakah anda ingin memesan lagi? (Y/N): ‘)

read (dlc.pilihan2)

}

if(dlc.pilihan2 == 'N'||dlc.pilihan2 == 'n'){

write ( ‘Terima Kasih sudah membeli’)

}

1. [Selesai]

Halt

B. Pointer

Algoritma Menghitung\_Pangkat

Algoritma untuk menghitung nilai yang dipangkatkan, dimana nilai dan pangkatnya diinput dengan angka yang diinginkan. Variabel \*pointx, \*pointy, hasil merupakan integer dimana \*pointx, \*pointy merupakan variabel pointer, variabel pointx dan pointy merupakan variabel new int (variabel baru).

[Deklarasi Pointer]

pointx = new int

pointy = new int

\*pointx = 0

\*pointy = 0

1. [Mulai]

[Masukkan nilai X (nilai yang akan dipangkatkan)]

Read (\*pointx)

1. [Masukkan nilai Y (nilai yang akan menjadi pangkatnya)]

Read (\*pointy)

1. [deklarasi dan menghitung rumus]

hasil = \*pointx \* \*pointx, \*pointy-1

1. [Tampilkan hasil akhir dari perhitungan program]

Write (\*pointx ‘ Dipangkatkan ‘ \*pointy ‘ = ‘ hasil)

1. [Selesai]

Halt

C. Rekursif

Algoritma Menghitung\_Pangkat

Algoritma untuk menghitung nilai yang dipangkatkan, dimana nilai dan pangkatnya diinput dengan angka yang diinginkan. long int p\_r merupakan fungsi rekursif, variabel x dan y bertipe integer.

[Deklarasi Fungsi Rekursif]

long int p\_r(int x, int y){

if (y==0){

return 1;

}

else{

return (x \* p\_r(x,y-1));

}

}

1. [Mulai]

[Masukkan nilai X (nilai yang akan dipangkatkan)]

Read (x)

1. [Masukkan nilai Y (nilai yang akan menjadi pangkatnya)]

Read (y)

1. [Tampilkan hasil akhir dari perhitungan program]

Write (x ‘ Dipangkatkan ‘ y ‘ = ‘ p\_r(x,y) )

1. [Selesai]

Halt