Function Kosong( Input list : Pointer ) 🡪 boolean

*{ I.S : penunjuk list pada stack sudah terdefinisi }*

*{ F.S : menghasilkan fungsi kosong}*

Kamus :

*{ tidak ada }*

Algoritma :

Kosong 🡨 false

If ( list = nil )

Then

Kosong 🡨 true

EndIf

EndFunction

Procedure Inisialisasi( Output Top : Pointer )

*{ I.S : memberi harga awal pada variabel penunjuk stack (Top) }*

*{ F.S : menghasilkan stack yang siap digunakan }*

Kamus :

*{ tidak ada }*

Algoritma :

Top 🡨 nil

EndProcedure

Procedure Isi( Output databaru : RIsi )

*{ I.S : user memasukkan data yang akan disisipkan (databaru) }*

*{ F.S : menghasilkan data yang akan disisipkan (databaru) }*

Kamus :

*{ tidak ada }*

Algoritma :

Output (' INPUT DATA PEMBELI ')

Output (' -------------------------------- ')

Output (' Nomor Kursi : ') Input ( databaru.nokursi )

Output (' Nama : ') Input ( databaru.nama )

Output (' Alamat : ') Input ( databaru.alamat )

Output (' No. HP : ') Input ( databaru.nohape )

Repeat

Output (' Kode Tujuan : ') Input( databaru.kodetujuan )

Until ( databaru.kodetujuan = 'CRB' ) or ( databaru.kodetujuan = 'BDG' ) or

( databaru.kodetujuan = 'JKT' )

If ( databaru.kodetujuan = 'CRB' )

Then

databaru.tujuan 🡨 'Cirebon'

Else

If ( databaru.kodetujuan = 'BDG' )

Then

databaru.tujuan 🡨 'Bandung'

Else

databaru.tujuan 🡨 'Jakarta'

EndIf

EndIf

Repeat

Output (' Kode Kelas : ') Input ( databaru.kodekelas )

Until ( databaru.kodekelas > 0 ) and ( databaru.kodekelas < 4 )

Case ( databaru.kodelas ) of

1 : databaru.kelas 🡨 'Eksekutif'

2 : databaru.kelas 🡨 'Bisnis'

3 : databaru.kelas 🡨 'Ekonomi'

EndCase

If ( databaru.kodekelas = 'CRB' )

Then

Case ( databaru.kodelas ) of

1 : databaru.harga 🡨 100000

2 : databaru.harga 🡨 80000

3 : databaru.harga 🡨 50000

EndCase

Else

If ( databaru.kodekelas = 'BDG' )

Then

Case ( databaru.kodelas ) of

1 : databaru.harga 🡨 150000

2 : databaru.harga 🡨 100000

3 : databaru.harga 🡨 80000

EndCase

Else

If ( databaru.kodekelas = 'JKT' )

Then

Case ( databaru.kodelas ) of

1 : databaru.harga 🡨 200000

2 : databaru.harga 🡨 150000

3 : databaru.harga 🡨 100000

EndCase

EndIf

EndIf

EndIf

EndProcedure

Procedure Output( Input data : RIsi )

*{ I.S : penunjuk data pada stack sudah terdefinisi }*

*{ F.S : menampilkan data }*

Kamus :

*{ tidak ada }*

Algoritma :

Output (' Nomor Kursi : ',data.nokursi )

Output (' Nama : ',data.nama )

Output (' Alamat : ',data.alamat )

Output (' No. HP : ',data.nohape )

Output (' Tujuan : ',data.tujuan )

Output (' Kelas : ',data.kelas )

Output (' Harga : ',data.harga )

EndProcedure

Procedure Push( I/O Top : Pointer, Input databaru : RIsi )

*{ I.S : data yang baru (databaru) dan pointer penunjuk stack (Top) sudah terdefinisi }*

*{ F.S : menghasilkan stack yang sudah ditambah sebuah data }*

Kamus :

baru : Pointer

Algoritma :

Alloc( baru )

baru↑.info 🡨 databaru

If ( Kosong( Top ) )

Then

baru↑.next 🡨 nil

Else

baru↑.next 🡨 Top

EndIf

Top 🡨 baru

EndProcedure

Procedure Pop( I/O Top : Pointer, Output elemen : RIsi )

*{ I.S : pointer penunjuk stack (Top) sudah terdefinisi }*

*{ F.S : menghasilkan stack yang sudah dikeluarkan sebuah datanya (elemen) }*

Kamus :

phapus : Pointer

Algoritma :

phapus 🡨 Top

elemen 🡨 phapus↑.info

If ( Kosong( Top↑.next ) )

Then

Top 🡨 nil

Else

Top 🡨 Top↑.next

EndIf

Dealloc( phapus )

EndProcedure

Procedure Tampil( I/O Top : Pointer )

*{ I.S : pointer penunjuk stack (Top) sudah terdefinisi }*

*{ F.S : menampilkan stack yang akan dihapus (elemen) }*

Kamus :

*{ tidak ada }*

Algoritma :

If ( not Kosong( Top ) )

Then

Repeat

Pop( Top, elemen )

Output( elemen )

Until ( Kosong( Top ) )

Else

Output (' Stack Kosong ')

EndIf

EndProcedure

*{ Algoritma Utama }*

Algoritma\_StackListTiket

Kamus :

*{ prototype }*

Function Kosong( Input list : Pointer ) 🡪 boolean

Procedure Inisialisasi( Output Top : Pointer )

Procedure Isi( Output databaru : RIsi )

Procedure Output( Input data : RIsi )

Procedure Push( I/O Top : Pointer, Input databaru : RIsi )

Procedure Pop( I/O Top : Pointer, Output elemen : RIsi )

Procedure Tampil( I/O Top : Pointer )

*{ deklarasi }*

Type

RecordIsi = Record

< nokursi : integer,

nama : string,

alamat : string,

nohape : string,

kodetujuan : string,

tujuan : string,

kodekelas : integer,

kelas : string,

harga : integer >

EndRecord

RIsi = RecordIsi

Pointer = ↑Stack

Stack = Record

< info : RIsi

next : Pointer >

EndRecord

Top : Pointer *{ pointer penunjuk }*

databaru : RIsi

elemen : RIsi

pil : integer

Algoritma :

Inisialisasi( Top )

Repeat

Output (' MENU PENJUALAN TIKET ')

Output (' KERETA API PT TORABIKA ')

Output (' -------------------------------------- ')

Output (' 1. Isi Data Pembeli ')

Output (' 2. Lihat Data Pembeli ')

Output (' 3. Keluar ')

Output (' -------------------------------------- ')

Output (' Pilihan = ') Input ( pil )

Case ( pil ) of

1 : Isi( databaru )

Push( Top, databaru )

2 : Tampil( Top )

EndCase

Until ( pil = 3 )