



LATIHAN STRUKTUR DATA: STACK (RAK BUKU)

Nama: Muhammad Ichsanul Fadhil

NIM: 103042400099

Soal 1: STACK

Suatu rak buku berdiri dapat menyimpan buku secara vertikal maksimum sebanyak 12 buku. Buku baru bisa disimpan ke dalam rak jika rak belum penuh dan akan disimpan di paling atas. Sebuah buku dapat diambil jika buku-buku yang di atasnya diambil terlebih dahulu, mulai dari buku yang paling atas.

a. Definisi Struktur Data (ADT)

Definisikan struktur data **RakBuku** menggunakan **Stack** yang menyimpan informasi judul buku dan penulis.

```
Type infoBuku : record {  
    judul    : string;  
    penulis  : string;  
}
```

```
Const MAX = 12
```

```
Type RakBuku : record {  
    data : array[1..MAX] of infoBuku;  
    top  : integer;  
}
```

Keterangan:

- MAX adalah kapasitas maksimum rak (12 buku).
- top menunjukkan posisi buku paling atas pada stack.

- Jika $top = 0$, maka rak kosong.

b. Prosedur Push (Menambah Buku ke Rak)

Menambahkan buku baru ke posisi paling atas jika rak belum penuh.

Procedure Push (InOut S : RakBuku, In judul, penulis : string)

{ IS: Stack S mungkin kosong.

FS: Buku baru menjadi elemen teratas pada stack S. }

Kamus Lokal:

B : infoBuku

Algoritma:

```

    if S.top < MAX then
        S.top ← S.top + 1
        B.judul ← judul
        B.penulis ← penulis
        S.data[S.top] ← B
    else
        output("Rak sudah penuh! Tidak dapat menambah buku.")
    end if

```

c. Prosedur Pop (Mengambil Buku Paling Atas)

Mengambil dan menghapus buku paling atas dari rak.

Procedure Pop (InOut S : RakBuku, Out judul, penulis : string)

{ IS: Stack S tidak kosong.

FS: Elemen teratas dihapus dan disimpan ke parameter output. }

Algoritma:

```

    if S.top > 0 then
        judul ← S.data[S.top].judul
        penulis ← S.data[S.top].penulis
        S.top ← S.top - 1
    end if

```

```

else
    output("Rak kosong! Tidak ada buku yang bisa diambil.")
end if

```

d. Prosedur Get (Mengambil Buku Berdasarkan Judul)

Mencari buku dengan judul tertentu. Buku-buku di atasnya akan dikeluarkan terlebih dahulu sampai buku yang dicari ditemukan.

```

Procedure Get (InOut S : RakBuku, In judul : string, Out namaPenulis : string)
{ IS: Stack S tidak kosong dan buku dengan judul tersebut ada di dalam rak.
  FS: Buku ditemukan, nama penulis disimpan ke variabel namaPenulis. }

```

Kamus Lokal:

```

temp : RakBuku
found : boolean
J, P : string

```

Algoritma:

```

CreateEmpty(temp)
found ← false

while S.top > 0 and found = false do
    Pop(S, J, P)
    if J = judul then
        namaPenulis ← P
        found ← true
    else
        Push(temp, J, P)
    end if
end while

while temp.top > 0 do
    Pop(temp, J, P)
    Push(S, J, P)
end while

```

```
if found = false then
    output("Buku dengan judul tersebut tidak ditemukan.")
end if
```

Penjelasan:

- Buku disimpan dan diambil sesuai prinsip *Last In First Out (LIFO)*.
- Prosedur **Get** menggunakan stack sementara untuk menjaga urutan buku setelah pencarian.
- Setelah buku ditemukan, isi rak dikembalikan seperti semula.

Dengan demikian, seluruh operasi **Push**, **Pop**, dan **Get** telah mengimplementasikan konsep dasar **Stack** untuk simulasi rak buku dengan kapasitas maksimum 12 elemen.