



# Praktikum Struktur Data

## Modul PSDA

### PSDA-11. ADT QUEUE – REPRESENTASI DENGAN LIST LINIER

1. Buatlah ADT Queue yang direpresentasikan dengan list linier dengan representasi fisik pointer dalam bahasa C sesuai dengan spesifikasi yang diberikan di bawah ini.
2. Buatlah driver untuk memeriksa apakah semua primitif sudah berjalan dengan baik.

```
{ Modul Queue }
{ Direpresentasikan dengan list linier, secara fisik dengan pointer }
{ Queue direpresentasikan sebagai list dengan first dan last }

constant Nil : ... { terdefinisi }

{ *** Definisi Type Queue *** }
type infotype : integer
type address : pointer to ElmtQueue type
ElmtQueue : < info : infotype,
              next : address >

{ Type Queue dengan ciri HEAD dan TAIL : }
type Queue : < HEAD : address, { alamat penghapusan } TAIL
              : address { alamat penambahan } >

{ *** Prototype manajemen memori *** }
procedure Alokasi (output P : address, input X : infotype)
{ I.S. P sembarang, X terdefinisi }
{ F.S. Alamat P dialokasi, jika berhasil maka Info(P) = X dan Next(P) = Nil }
{ P = Nil jika alokasi gagal }
procedure Dealokasi (input/output P : address)
{ I.S. P adalah hasil alokasi, P <> Nil }
{ F.S. Alamat P didealokasi, dikembalikan ke sistem }

{ *** Predikat Pemeriksaan Kondisi Queue *** }
function IsEmpty (Q : Queue) → boolean
{ Mengirim true jika Q kosong: HEAD(Q) = Nil and TAIL(Q) = Nil }
function NBElt (Q : Queue) → integer
{ Mengirimkan banyaknya elemen queue. Mengirimkan 0 jika Q kosong. }

{ *** Konstruktor *** }
procedure CreateEmpty (input/output Q : Queue)
{ I.S. sembarang }
{ F.S. Sebuah Q kosong terbentuk }

{ *** Primitif Add/Delete *** }
procedure Add (input/output Q : Queue, input X : infotype)
{ Proses : Mengalokasi X dan menambahkan X pada bagian TAIL dari Q jika alokasi
  berhasil; jika alokasi gagal Q tetap }
{ I.S. Q mungkin kosong }
{ F.S. X menjadi TAIL, TAIL "maju" }
procedure Del (input/output Q : Queue, output X : infotype)
{ Proses : Menghapus X pada bagian HEAD dari Q dan mendealokasi elemen HEAD }
{ I.S. Q tidak mungkin kosong }
{ F.S. X = nilai elemen HEAD pd I.S., HEAD "mundur" }
```