

JURNAL Modul 11

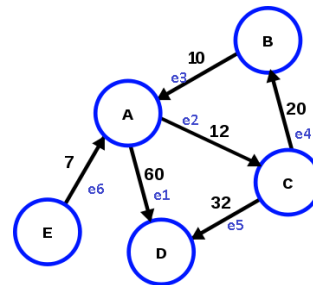
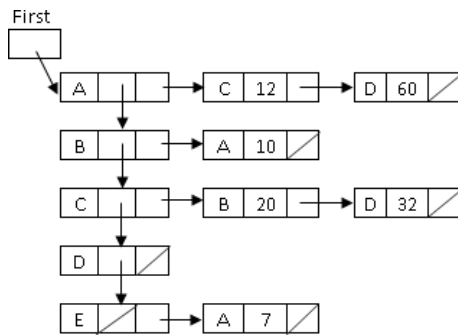
STRUKTUR DATA - Ganjil 2024/2025

"Graph"

A. Tugas Pendahuluan

NOTE : SECTION INI BAGIAN YANG SAMA DENGAN TUGAS PENDAHULUAN SEBELUMNYA,

Sebuah **graf berarah dan berbobot** direpresentasikan dengan multi linked-list berikut:



ADT

```

type adrVertex: pointer to vertex
type adrEdge: pointer to edge
type vertex: <idVertex: char,
              nextVertex: adrVertex,
              firstEdge: adrEdge >
type edge: <destVertexID: char,
            weight: integer,
            nextEdge: adrEdge>
type graph <firstVertex: adrVertex>
  
```

Buat procedure dan fungsi berikut berikut :

procedure createVertex (input newVertexID: char, output v:adrVertex)

{I.S. Terdefinisi ID untuk vertex/simpul baru

F.S. Telah dialokasi sebuah simpul baru v dengan idVertex(v)=newVertexID, nextVertex(v)=nil, firstEdge(v)=nil}

procedure initGraph (output G: graph)

{I.S. -

F.S. terdefinisi sebuah graph G dimana firstVertex(G)=nil}

procedure addVertex (input/output G: graph, input newVertexID: char)

{I.S. Terdefinisi sebuah graf, G, dan sebuah ID vertex/simpul yang akan ditambahkan ke dalam G.

F.S. Simpul baru dengna ID=newVertexID sudah ditambahkan ke dalam graf G}

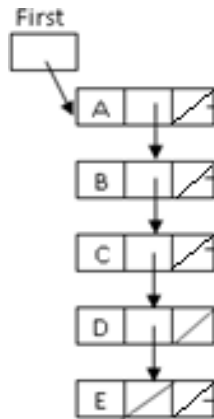
procedure buildGraph (output G: graph)

{I.S.-

Proses: menerima ID vertex/simpul baru dari piranti masukan, kemudian menambahkan simpul baru itu ke dalam graf G. ID vertex adalah huruf (A..Z). Proses dilakukan berulang sampai diterima character yang bukan huruf.

F.S. Terbentuk sebuah graf G yang setiap simpulnya mempunyai ID yang unik (tidak ada yang sama satu sama lain. Graf G belum mempunyai busur }

Untuk graf di atas, hasil dari prosedur buildGraph sebagai berikut:



B. Tugas Jurnal

NOTE : SILAHKAN KERJAKAN SECTION INI UNTUK Pengerjaan Jurnal

Anda sudah mengerjakan tugas pendahuluan (soal test pendahuluan dicantumkan di halaman 2). Lanjutkan dengan membuat fungsi/prosedur di bawah ini:

function searchVertex (G:graph, IDsimpul:char) → adrVertex

{Fungsi akan mencari vertex yang berisi IDsimpul. Fungsi akan mengirimkan alamat elemen vertex tersebut jika ditemukan atau NIL jika tidak ditemukan.}

procedure createEdge (input IDsimpulTujuan: char, input bobot:integer, output e:adrEdge)

{I.S. Terdefinisi ID simpul tujuan

F.S. Telah dialokasi sebuah elemen edge baru bernama e, dimana destVertexID(e)=IDsimpulTujuan, weight(e)= bobot, dan nextEdge (e)=nil}

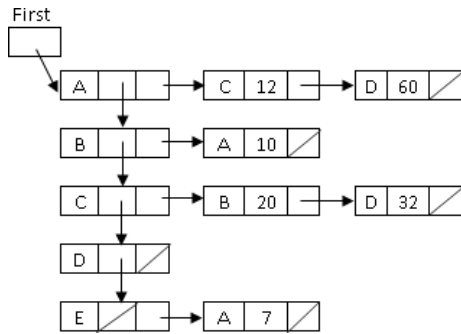
procedure addEdge (input/output G: graph, input IDsimpulAsal, IDsimpulTujuan: char, bobot:integer)

{I.S. Terdefinisi sebuah graf, G. Juga diketahui ID simpul asal dan tujuan sebuah edge beserta bobotnya.

Proses: Cari elemen vertex dengan ID=IDsimpulAsal. Kemudian, tambahkan busur baru yang menghubungkan simpul asal dan simpul tujuan.

F.S. Sebuah edge baru dari telah ditambahkan, dimana destVertexID(e)=IDsimpulTujuan, weight(e)= bobot, dan nextEdge (e)=nil}}

Panggil addEdge berulang kali sehingga terbentuk representasi graf di bawah ini:



function outDegree (G:graph, IDsimpul:char) → integer

{ Fungsi akan mengirimkan derajat keluar (out-degree) untuk simpul dengan ID = IDsimpul }

function inDegree (G:graph, IDsimpul:char) → integer

{ Fungsi akan mengirimkan derajat masuk (in-degree) untuk simpul dengan ID = IDsimpul }

function degree (G:graph, IDsimpul:char) → integer

{ Fungsi akan mengirimkan derajat untuk simpul dengan ID = IDsimpul }