

TUGAS STRUKTUR DATA DAN ALGORITMA
STUDI KASUS ADT LINKED LIST



Oleh :
ARSAL FADILAH
201524036

POLITEKNIK NEGERI BANDUNG
JURUSAN TEKNIK KOMPUTER
D-4 TEKNIK INFORMATIKA
2021

- Pseudocode

Struktur data :

```
typedef char infotype;
typedef struct tElmtlist *address;
typedef struct tElmtlist {
    infotype info;
    address next;
} ElmtList;
```

```
typedef struct {
    address First;
} List;
```

- Create List

Deklarasi :

-

Algoritma :

Procedure CreateList(Input/Output List L)

Begin

L^.First = Nil

End

- Insert Huruf Pertama, value 'u'

Deklarasi :

Address P

Algoritma :

Procedure InsertFirst(Input. Output List L, Infotype X)

Begin

P ← alloc(1, address)

If P != Nil then

P^.Info ← X

P^.Next ← L.First

L.First ← P

End If

End

- Insert value 'c' setelah huruf pertama

Deklarasi :

Address P, Last

Algoritma :

Procedure InsertLast(Input. Output List L, Infotype X)

Begin

P ← alloc(1, address)

If P != Nil then

P^.Info ← X

If L.First == Nil then

P^.Next ← L.First

L.First ← P

Else then

Last ← L.First

While Last^.Next != Nil

Last ← Last^.Next

```

                                End While
                                Last^.Next ← P
                                End If
                            End If
                        End
○ Insert value 'h' setelah huruf kedua
Deklarasi :
Addres P, Last
Algoritma :
Procedure InsertLast(Input.Output List L, Infotype X)
Begin
    P ← alloc(1, address)
    If P != Nil then
        P^.Info ← X
        If L.First == Nil then
            P^.Next ← L.First
            L.First ← P
        Else then
            Last ← L.First
            While Last^.Next != Nil
                Last ← Last^.Next
            End While
            Last^.Next ← P
        End If
    End If
End
○ Insert value 'i' setelah huruf ketiga
Deklarasi :
Addres P, Last
Algoritma :
Procedure InsertLast(Input.Output List L, Infotype X)
Begin
    P ← alloc(1, address)
    If P != Nil then
        P^.Info ← X
        If L.First == Nil then
            P^.Next ← L.First
            L.First ← P
        Else then
            Last ← L.First
            While Last^.Next != Nil
                Last ← Last^.Next
            End While
            Last^.Next ← P
        End If
    End If
End

```

- Insert value 'n' setelah huruf pertama

Deklarasi :

-

Algoritma :

Procedure InerAfter(Input.Output List L, address P, address Prec)

Begin

$P^{\wedge}.Next \leftarrow Prec^{\wedge}.Next$

$Prec^{\wedge}.Next \leftarrow P$

End

- Delete value 'i' pada elemen list

Deklarasi :

Address Last, PrecLast

Algoritma :

Procedure DeleteLast(Input/Output List L, input/output address P)

Begin

$Last \leftarrow *(L.First)$

 If $Last^{\wedge}.Next == Nil$ then

$Last \leftarrow *(L.First)$

$*(L.First) \leftarrow *(L.First)^{\wedge}.Next$

$Last.Next \leftarrow Nil$

 Else then

$PrecLast \leftarrow Nil$

 While $Last^{\wedge}.Next \neq Nil$ then

$PrecLast \leftarrow Last$

$Last \leftarrow Last^{\wedge}.Next$

 End While

$*P \leftarrow Last$

$PrecLast^{\wedge}.Next \leftarrow Nil$

 End If

End

- Insert value 'p' di awal list Insert value 'a' setelah huruf 'h'

Deklarasi :

-

Algoritma :

Procedure InerAfter(Input.Output List L, address P, address Prec)

Begin

$P^{\wedge}.Next \leftarrow Prec^{\wedge}.Next$

$Prec^{\wedge}.Next \leftarrow P$

End

- Show semua elemen karakter

Deklarasi :

Addres P

Algoritma :

Procedure PrintInfo(Input List L)

```

Begin
    P ← L^.First
    If P == Nil then
        Write (layar) "List Kosong"
    Else
        Do then
            Write (layar) P^.info
            P = P^.Next
        While P != Nil
    End If
End

```

- Cari huruf 'h' pada list, dan tampilkan posisi elemen tersebut pada list. Jika ada tampilkan posisinya, jika tidak ada tampilkan "tidak ada"

Deklarasi :

Int idx ← 1

Algoritma :

Function IdxNList(Input List L, infotype x) : int

```

Begin
    While L.First != Nil And L.First^.Info != x then
        Idx++
        L.First ← L.First^.Next
    End while
    If L.First^.info != x then
        Return 0
    Else
        Return idx
    End if
End

```

- Delete semua elemen list karakter

Deklarasi :

Address X

Algoritma :

Procedure DelAll(Input/Output List L)

```

Begin
    While L.First != Nil then
        DeleteLast(L, X)
    End while
End

```

- Representasi/Illustrasi Gambar Linked List

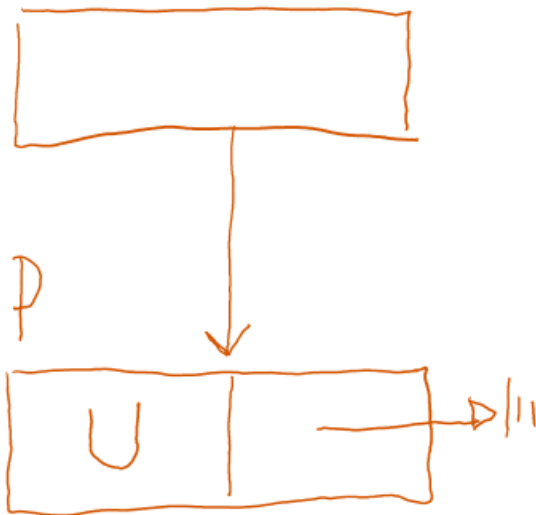
- Create List

L, First



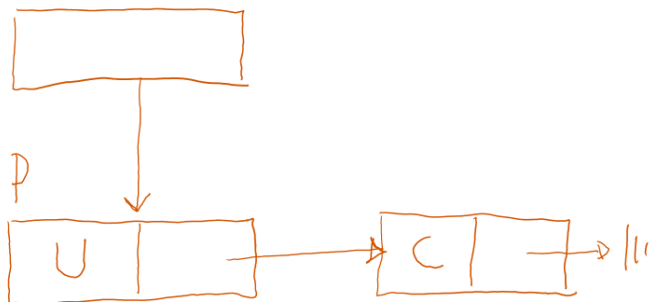
- Insert Huruf Pertama, value 'u'

L, First

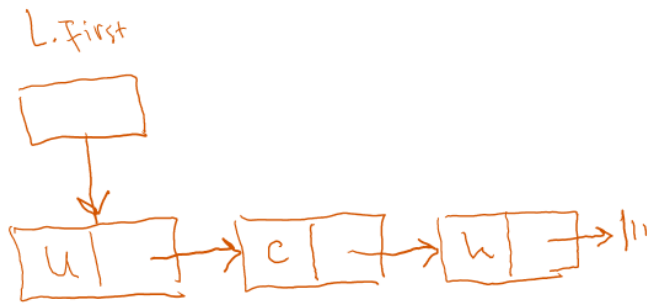


- Insert value 'c' setelah huruf pertama

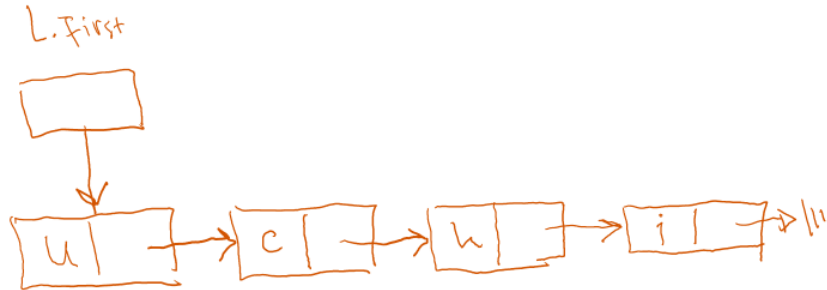
L, First



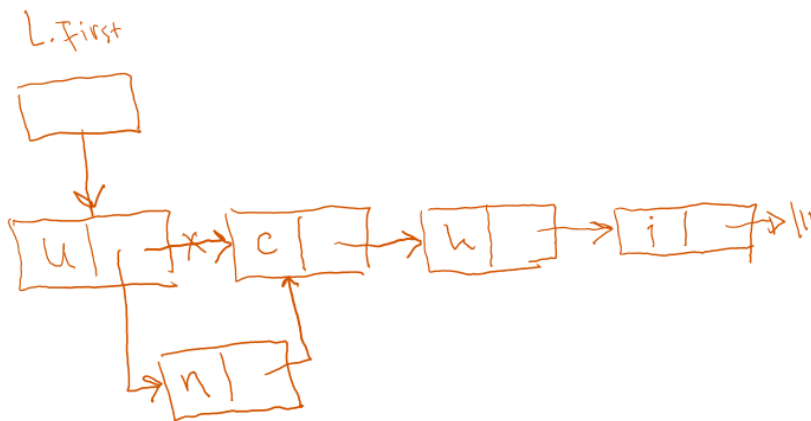
- Insert value 'h' setelah huruf kedua



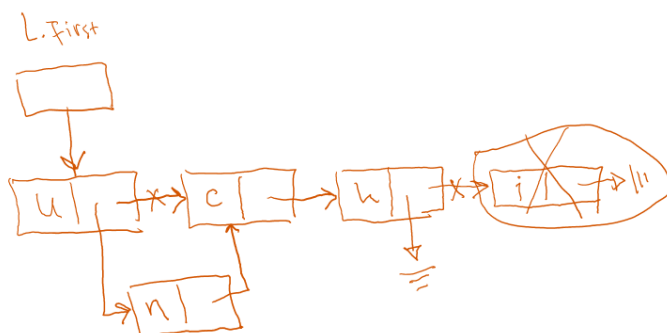
- Insert value 'i' setelah huruf ketiga



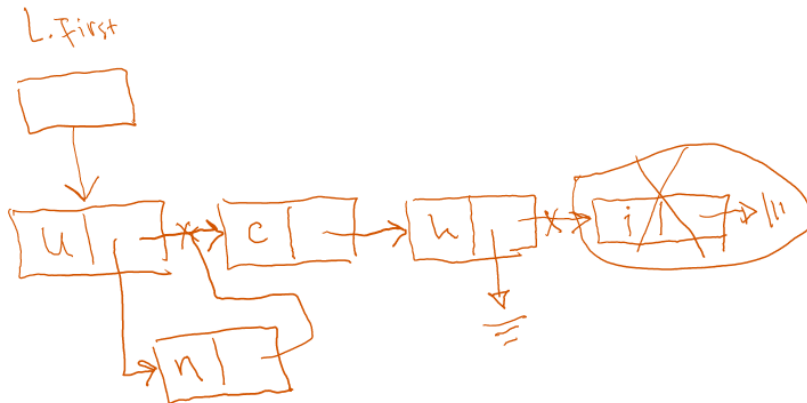
- Insert value 'n' setelah huruf pertama



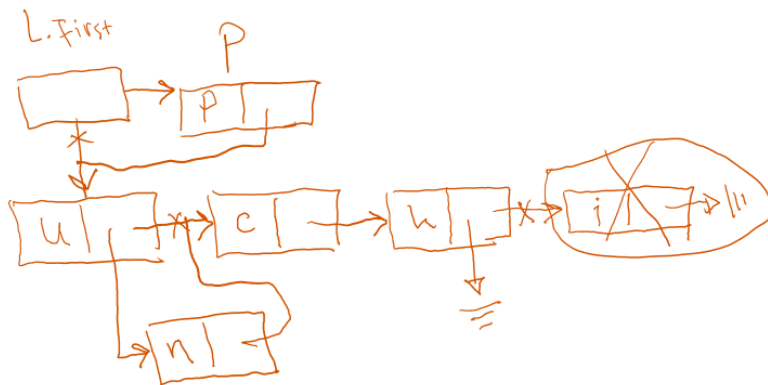
- Delete value 'i' pada elemen list



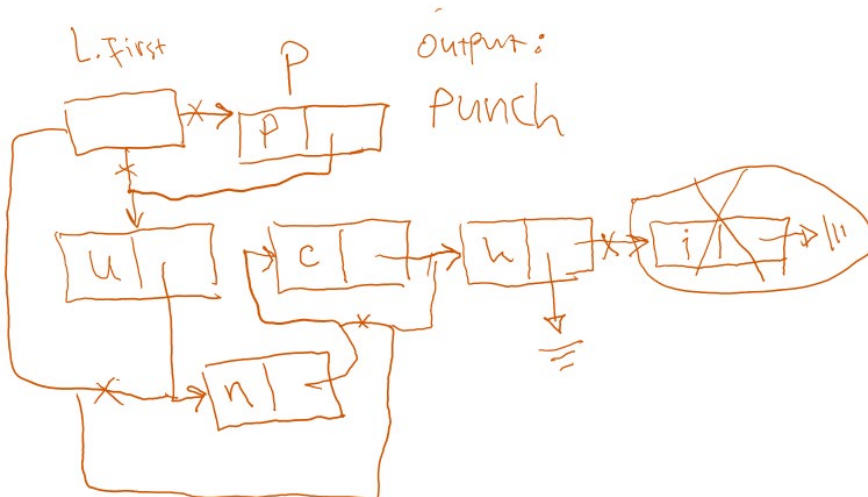
- Insert value 'p' di awal list



- Insert value 'a' setelah huruf 'h'

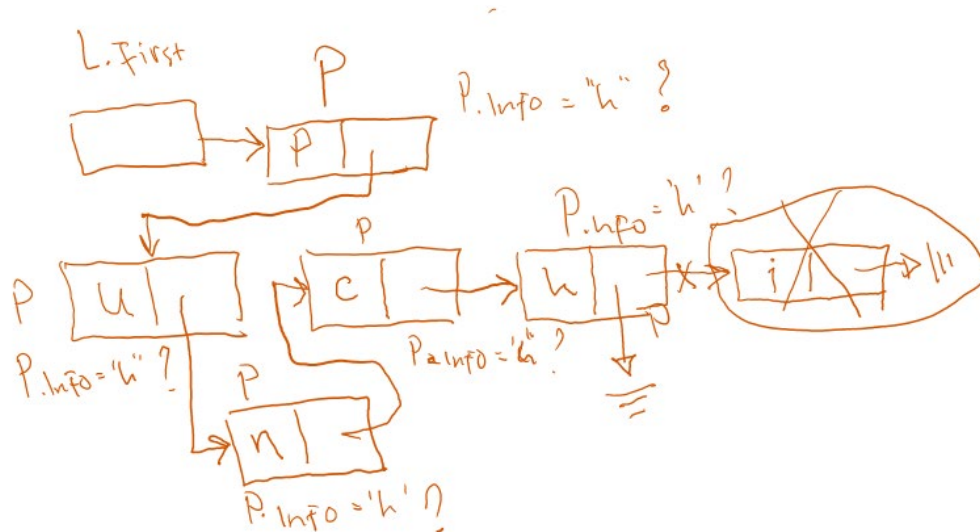


- Show semua elemen karakter

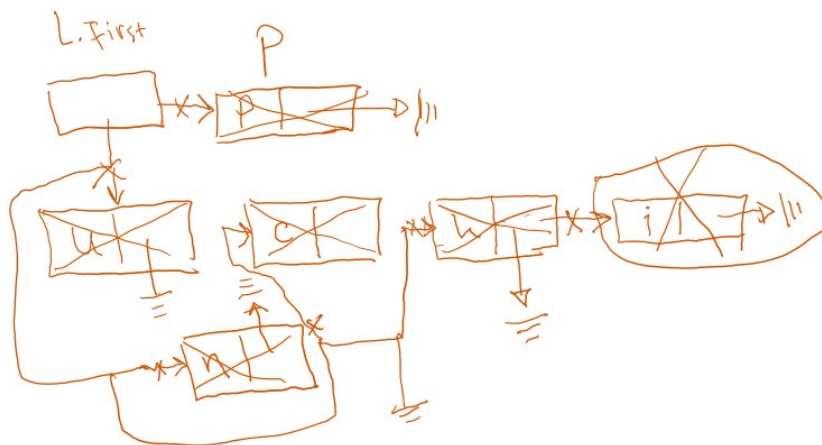


Cat : L.First sebagai parameter input saja

- Cari huruf 'h' pada list, dan tampilkan posisi elemen tersebut pada list. Jika ada tampilkan posisinya, jika tidak ada tampilkan "tidak ada"



○ Delete semua elemen list karakter



//Hasil Running

```
E:\Arsal\Materi Kuliah\Struktur Data dan Algoritma\Praktek\Tugas Linked List\LinkList (Praktek)>AppChar
Create List
Insert huruf pertama, value 'u'
Print List ->> [ u ]

Insert value 'c' setelah huruf pertama
Print List ->> [ u c ]

Insert value 'h' setelah huruf kedua
Print List ->> [ u c h ]

Insert value 'i' setelah huruf ketiga
Print List ->> [ u c h i ]

Insert value 'n' setelah huruf pertama
Print List ->> [ u n c h i ]

delete value 'i' pada elemen list
Print List ->> [ u n c h ]

Insert value 'p' di awal list
Print List ->> [ p u n c h ]

Insert value 'a' setelah huruf 'h'
Print List ->> [ p u n c h a ]

Show semua elemen karakter
[ p u n c h a ]

Cari huruf 'h' pada list
Posisi ke 5 dengan alamat 181420

Delete semua elemen list karakter
Print List ->> List Kosong !
```