Fakultas Informatika
School of Computing
Telkom University

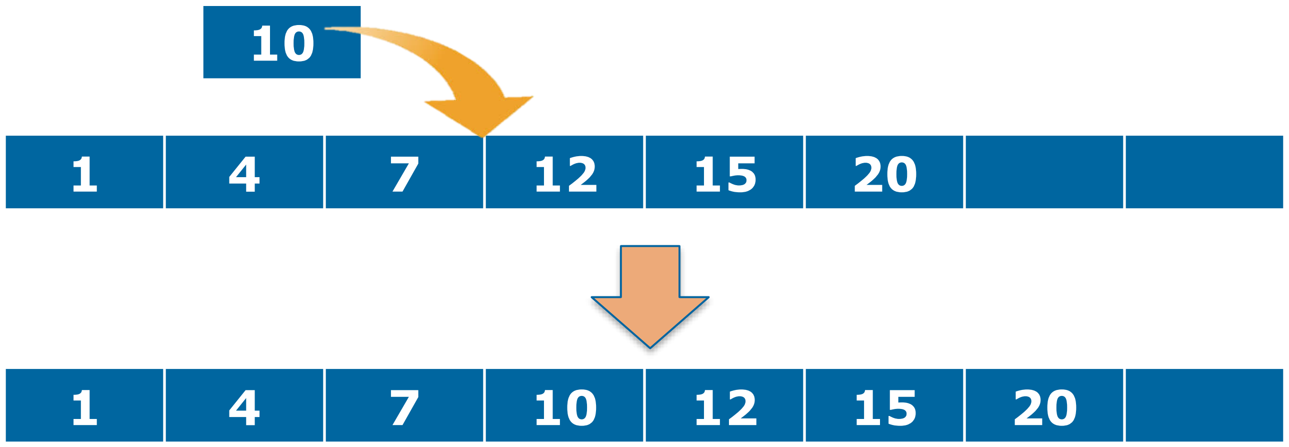
CCK2AAB4
STRUKTUR DATA

Single Linked List

Introduction

Exercise

- Create an algorithm to insert a number into an ordered array of integer so that the array result remain ordered

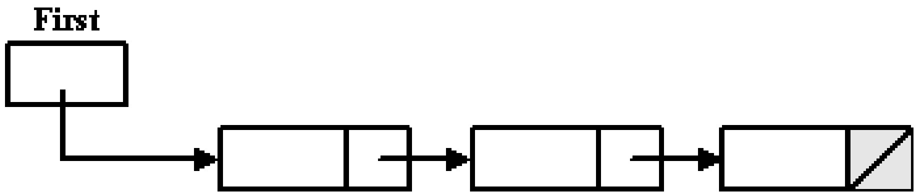


Insert into a sorted Array

```
Algorithm
    while ( i < n ) and ( tab[i] < x ) do
        i++
    temp1 ← tab[i]
    tab[i] ← x
    j traversal [i+1..n]
        temp2 ← tab[j]
        tab[j] ← temp1
        temp1 ← temp2
```

Troublesome isn’t it?

- that’s why we learn about Linked List
- Dynamic Array

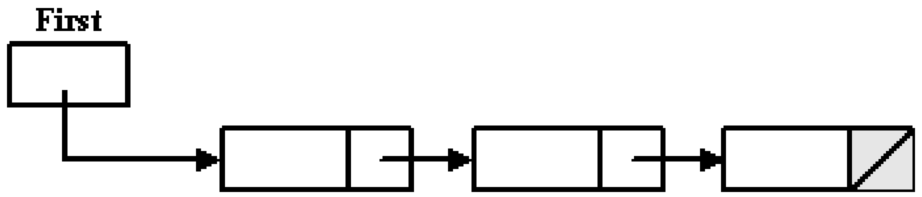


Linked List

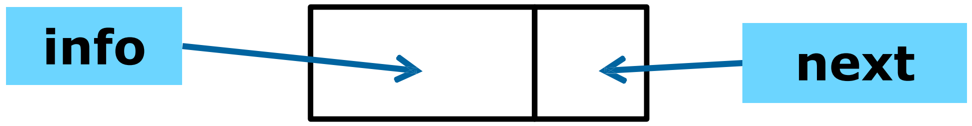
- a data structure in which each element is allocated dynamically and are bound with other elements to form a linear relationship
- This structure allows for efficient insertion or removal of elements from any position in the sequence

Structure

- Consists of nodes/elements

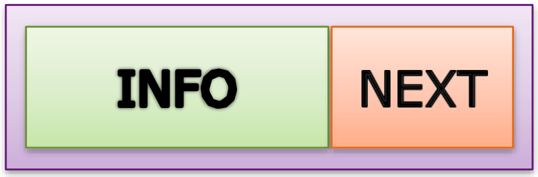


- Generally, each Element is divided into 2 parts



Element List

```
Type ElmList <
    info : infotype
    next : address
>
```



ElmList

What is infotype ?

What is address ?

Infotype

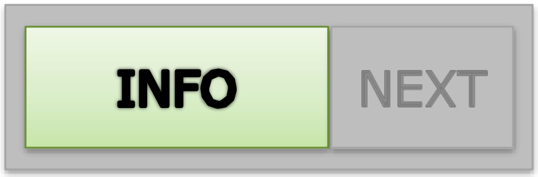
- The data that we want to store
- Define your own infotype
  - Basic type example

    Type infotype : integer

    Type infotype : char

  - Record type example

```
Type infotype :
mahasiswa <
        nim : string
        name : string
>
```

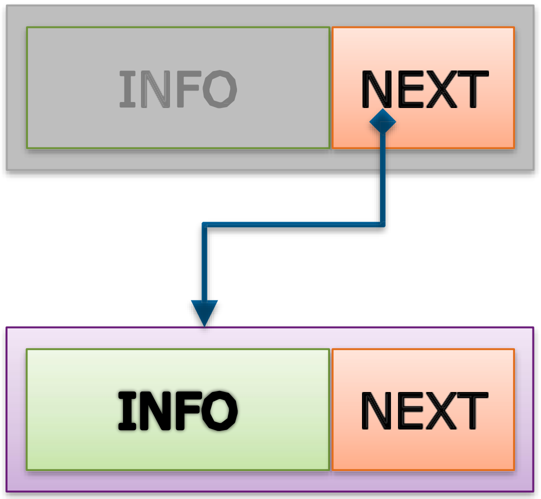


ElmList

Address

- Pointer to element

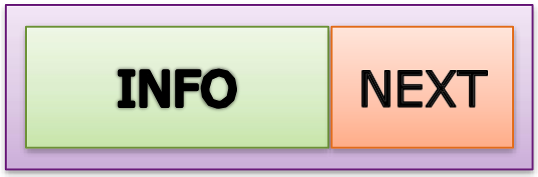
  Type address : pointer to ElmList



ADT Element List

```
Type infotype : integer
Type address : pointer to ElmList

Type ElmList <
    info : infotype
    next : address
>
```

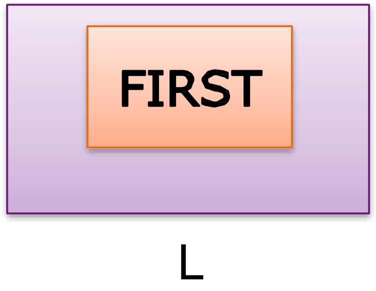


ElmList

Single Linked List

```
Type List : < First : address >

Dictionary
    L : List
```

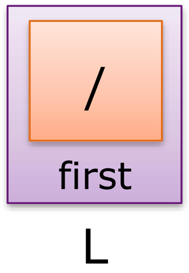


- Only create the list variable

Create New List

Algorithm

```
    First(L) ← Nil
```

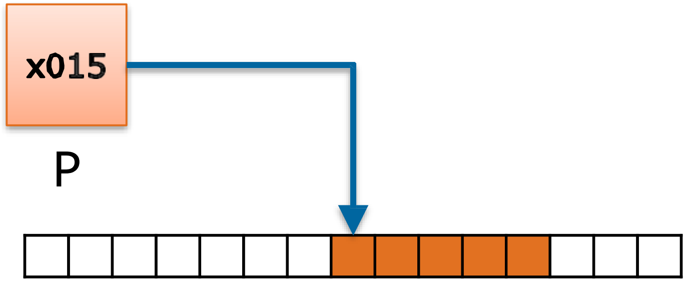


- **First(X)** is a keyword to know the first element of the list X
  - Use First(X) instead of X.First
- On the creation of new list, there is no element, thus first(L) is Nil / Null

Creating New Element

Algorithm

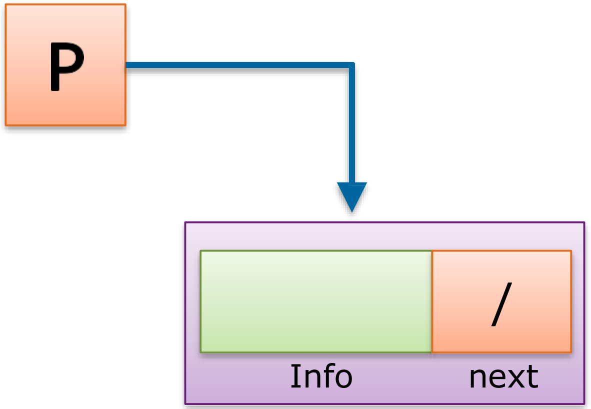
```
    Allocate(P)
```



- Allocating space memory for an element
  - According to the size defined by the element type
- Only the pointer that knows where the element resides

Creating New Element

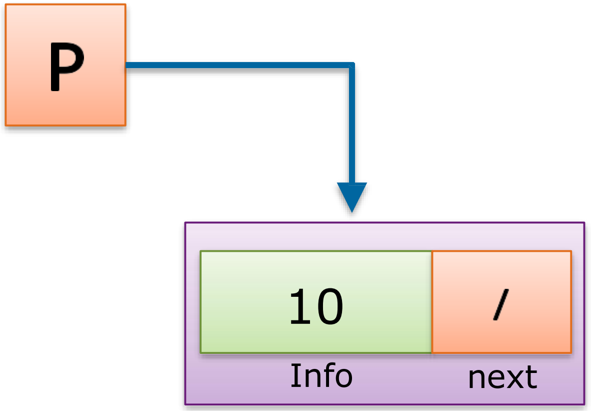
```
Algorithm
    Allocate(P)
    Next(P) ← Null
```



- **Next(Y)** is a keyword to know the next element of element pointed by Y
  - Likewise, use Next(Y) instead Y.Next
- On the creation of new element, set Next element = Null

Creating New Element

```
Algorithm
    Allocate(P)
    Next(P) ← Null
    Info(P) ← 10
```

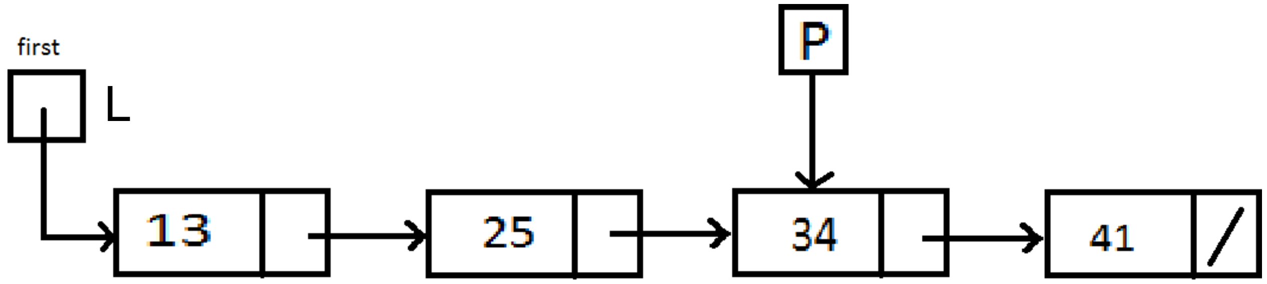


- **Info(Y)** is a keyword to access the data stored in the element
  - If infotype is a record type, operation is like a normal record operation
  - Info(P).nim ← '11031300xx'

Keywords

- **First(X)**
  - Select the first element of list X
- **Next(Y)**
  - Select the next element of element Y
- **Info(Y)**
  - Select the data stored in element Y

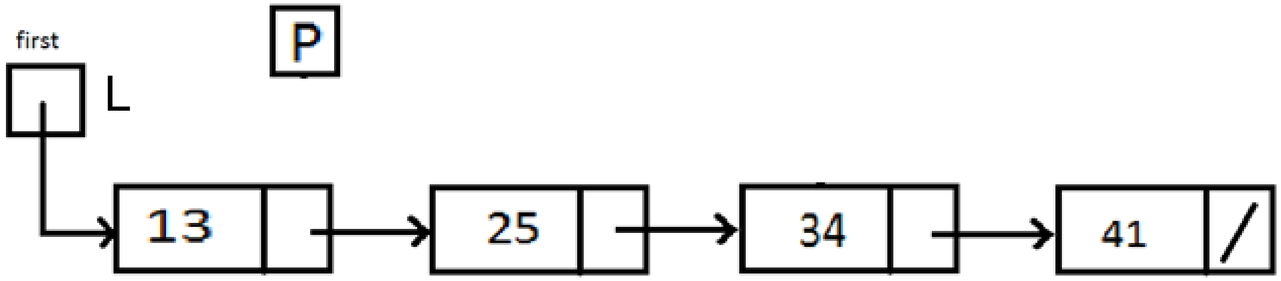
Exercise



Task	Answer
Output(P.info)	
Output((L.first).info)	
Output((P.next).info)	

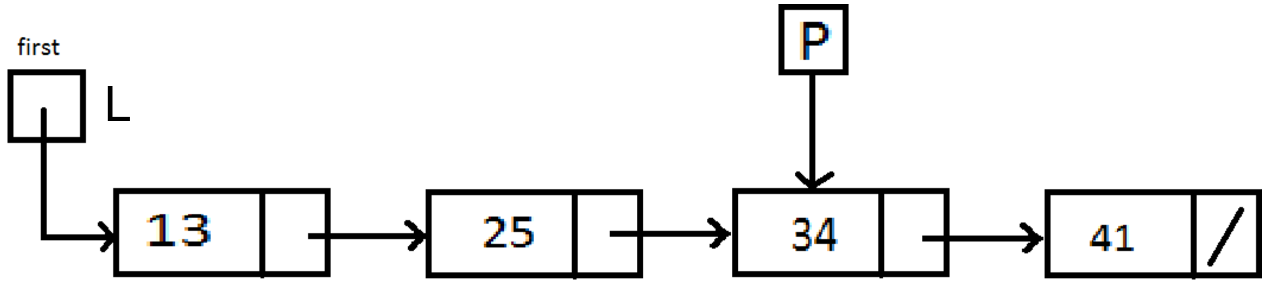
P ← (L.first).next
Output((P.next).info)	

Exercise



Task	Answer
Make P points the first element	
Make P points the second element	
Make P points the last element	
Output info the first element of the list	
Output info of the last element	

Exercise



Task	Answer
Copy info element P into first element	
Copy info the second element into P	
Set info of first element = 10	
Output info element P	
Output info of first element	
Copy info first element into next element of P	
Output info of the last element	

Question?

Fakultas Informatika
School of Computing
Telkom University

THANK YOU