

Data Structures

Agar mengetahui struktur data

No. _____

Date _____

Pointer

When a variable is declared, a specific block of memory within the computer is allocated to hold the value (the size depends of the data type)

In 64 bit system, 8 bytes

In 32 bit system, 4 bytes

In 16 bit system, 2 bytes

Setiap compiler memiliki cara akses memori yg berbeda

Two most important operator used with pointer type are :

& the address operator

* the dereferencing operator

scanf hrs take & karena untuk memberi tau compiler untuk mengetahui dimana variable itu berada.

Pointer

- Use to pass information back and forth between a function and its reference point.
- Use to create complex data structures, such as trees, linked list, linked stack, linked queue and graphs.
- Use for the dynamic memory allocation of variable
- Provide an alternative way to access the individual elements of an array
- Enable the programmers to return multiple data items from a function via a function arguments or to pass arrays and strings as function arguments

Ex:

```
int x;
```

```
int *px;
```

$px = 8x$; \rightarrow mengkas: memberi tau baru nilai apapun yg diberi ke px menjadi nilai x

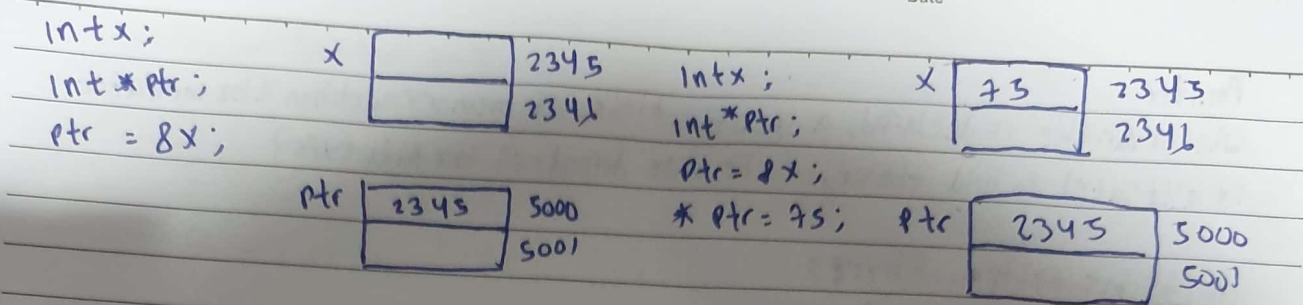
```
x = 10;
```

```
*px = 20;
```

```
printf (" * %d \n", x);
```

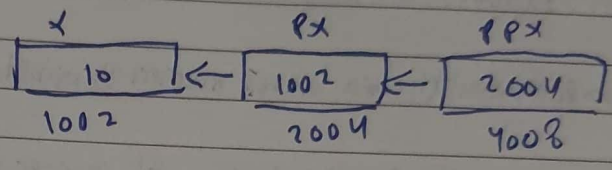
Ans: x=20.

estudee



Pointer to Pointer

```
Ex: int x = 10;
int *px, **ppx;
px = &x;
ppx = &px;
```



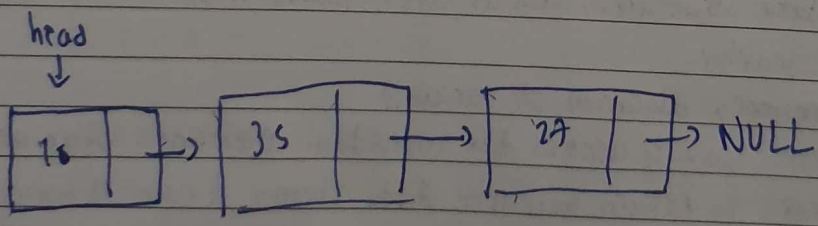
Array

~~Linear list is an~~

A collection of similar data ~~type~~ elements
Array index starts from zero.

↳ Karena kalau mulai dari 1 binarynya tidak efisien.

Linked list



- Example of a list whose nodes contain two fields :
 - An integer value and a link to the next node.
 - Linked list which node contain only a single link to other node is called single linked list.

Note: Linked list vs Array

Array:

- Linear collection of data elements
- store value in consecutive memory locations.
- Can be random in accessing of data

Linked list:

- Linear collection of nodes.
- Doesn't store its nodes in consecutive memory locations
- Can be ~~asse~~ accessed only in sequential manner

Memory Allocation: Dynamic

- If you need to allocate memory dynamically, you can use malloc in C/C++.
- To de-allocate you can use free.

malloc returns point *

Example:

```
int * px = (int *) malloc (size of (int));
```

```
char * pc = (char *) malloc (size of (char));
```

```
* px = 205;
```

```
* pc = 'A';
```

```
printf ("%d %c\n", *px, *pc);
```

```
free (px);
```

```
free (pc);
```

Single linked list

- To create a linked list, we first need to define a node structure for the list.
- Supposed we want to create a list of integers.

Ex:-

```
struct Tnode {
```

```
    int value;
```

```
    struct Tnode * next;
```

```
};
```

```
struct Tnode * head = 0;
```

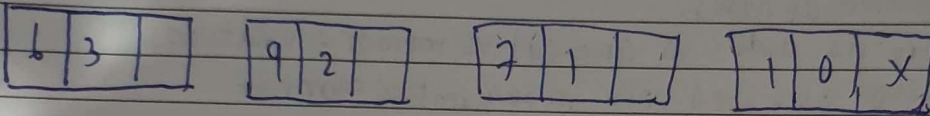
estudee

24 Feb 2020

No.
Date

Polynomial Representation

- Polynomial is given as $6x^3 + 9x^2 + 7x + 1$
- Every individual term in a polynomial consists of two parts, a coefficient and a power.
- Here 6, 9, 7, and 1 are the coefficients of the ~~ter~~ terms that have 3, 2, 1, and 0 as their power respectively.
- Every term of polynomial can be represented as a node of the linked list.

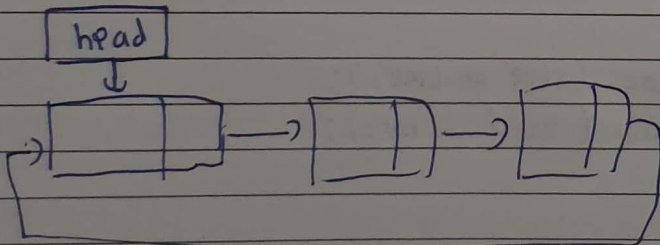


Pertemuan 2

Linked list II (2)

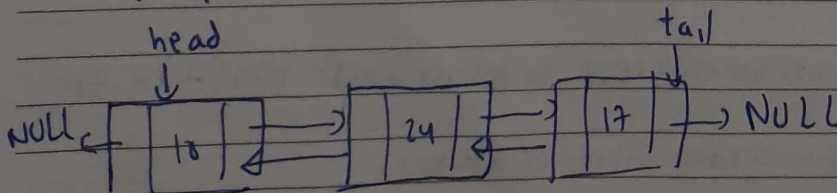
Circular single linked list

- In circular, last node contains a pointer to the first node



Double linked list / two way linked list

- ~~It~~ is a linked list data structure with two links, one that contains reference to the next data and one that contains reference to the previous



```
struct tnode {
```

```
int value;
```

```
struct tnode * next;
```

```
struct tnode * prev;
```

```
};
```

```
struct tnode * head = 0;
```

```
struct tnode * tail = 0;
```

estudee

Doubly linked list : Insert

- We should allocate the new node and assign the value to it, and then we connect the node with the existing linked list.
- Supposed we want to append the new node behind the tail.

```
struct tnode *node  
(struct tnode *) malloc (sizeof (struct tnode));  
node -> value = x ;  
node -> next = NULL ;  
node -> prev = tail ;  
tail -> next = node ;  
tail = node ;
```

Stack and Queue

Stack concept

- Stack is an important data structure which stores its elements in an ordered manner.
- Stack is a linear data structure which can be implemented by either using an array or a linked list.

Stack Operation

- Push(x) : add an item x to the top of the stack.
- pop() : remove an item from the top of the stack.
- top() : reveal / return the top item from the stack.