Whenever a value is placed in a memory location, the value replaces the previous value in that location and the previous value is lost; thus, this process is said **to be destructive**.

If an expression contains several multiplication, division and remainder operations, evaluation proceeds from left to right. **Multiplication, division and remainder are said to be on the same level of precedence**

Hvordan en funktion der tager en arrays som variable

Hvordan overføres en array by reference or by copy

Hvad er en pointer

Hvordan benyttes en pointer ved funktionskald

Hvordan kaldes en funktion med en parameter,

Hvordan kaldes en funktion hvis der ønskes overført en n erklæres en funktion med en variable som parameter

Hvad er alternav til en for løkke

Hvordan erklæres en for løkke

Hvordan erklæres en while løkke

Hvordan erklæres en do while løkke

Hvad bruger man løkker til giv eksempler fra opgaverne

Switch case - hvordan ser det ud

Array hvordan erklæres en char array

Hvordan benyttes array som parametre ved funktions kald

Hvilke to måder er der at overføre variable til funktioner på?

Pointer erklæringer

Use a Pointer Parameter to Receive an Address **A function receiving an address as an argument must define a pointer parameter to receive**

The Indirection (*) Operator The unary * operator, commonly referred to as the indirection operator or **dereferencing operator**, returns the value of the object to which its operand

D=*ptrV

There are four ways to pass a pointer to a function:

• a non-constant pointer to non-constant data. Void enfunktion (char *charPtr)

• a constant pointer to nonconstant data. Void enfunktion(char *const charPtr) int * const ptr = &x;

• a non-constant pointer to constant data. Void enfunktion(const char *charPtr) - f.eks. ved udskrift a fen string

• a constant pointer to constant data. const int *const ptr = &x;

Arrays of pointers const char *suit[4] = {"Hearts", "Diamonds", "Clubs", "Spades"};

Hvordan initialiseres en array til en string

It's possible to **pass an array by value (by placing it in a struct as** we explain in Chapter 10,

static local variable exists for the duration of the program but is visible only in the function body

hvad betyder det når man skriver **const** for an type erklæringen for en array?

Giv et eksemple på en multi-dimentionel array - række søjle

Structures may be passed to functions by

• passing individual structure members.

• passing an entire structure. By copy of ex. array

• passing a pointer to a structure

typedef struct card **Card; ny type**

As with a struct definition, a union definition simply creates a new type

Placing a union or struct definition outside any function does not create a global variable.

—**so a union shares the space** instead of wasting storage on variables that are not being used.

Relationship between FILE Pointers, FILE Structures and FCB a file control block (FCB)—

a file for sequential access

Data in this type of sequential file cannot be modified without the risk of destroying other data.

Therefore, sequential access with fprintf and fscanf is not usually used to update records in place. Instead, the entire file is usually rewritten.

introduce random-access files Raw data - wb

However, individual records that you write to and read from a random-access file are normally fixed in length and may be accessed directly (and thus quickly) without searching through other records

Datastrukturer

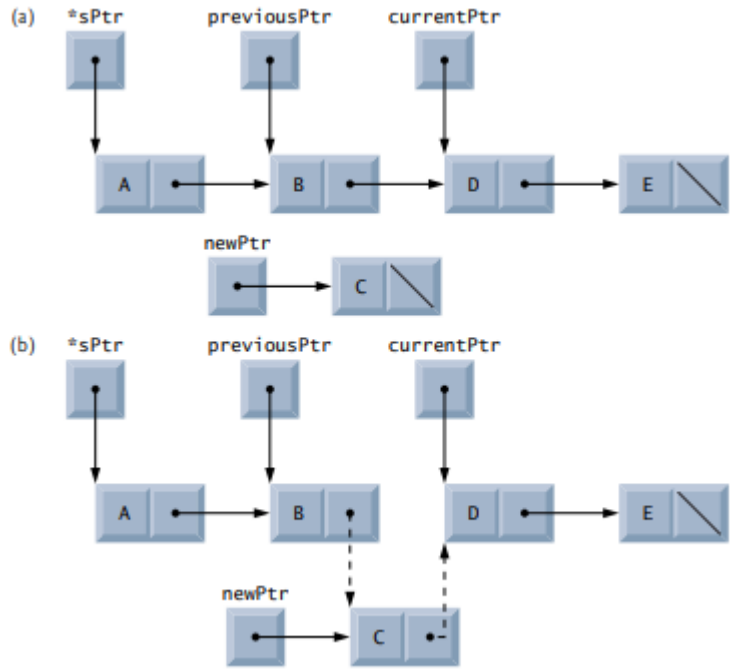
Hægtede lister

Kø FiFO

Stack LIFO

Hvordan ser grund element ud I en liste

```
struct listNode {
        char data; // each listNode contains a character
        struct listNode *nextPtr; // pointer to next node
};
```



Pointers previousPtr and currentPtr store the locations of the node preceding and after the insertion point, respectively.

JSON formats

Data is in key-value pairs.

{ "name": "John", "age": 30 }

Data is separated by commas.

{ "name": "John", "age": 30, "city": "New York" }

Curly braces {} hold objects.

Ex: "wind": {

"speed": 7.94,

"deg": 64,

"gust": 10.98

},

Square brackets [] hold arrays.

{ "fruits": ["apple", "banana", "cherry"] }

Data types:

- String: Text enclosed in double quotes.
  - { "message": "Hello, World!" }
- Number: Integers or floating-point.
  - { "price": 19.99 }
- Boolean: true or false.
  - { "inStock": true }
- Array: Ordered list of values.
  - { "colors": ["red", "green", "blue"] }
- Object: Unordered collection of key-value pairs.
  - { "user": { "id": 1, "name": "Alice" } }
- Null: Empty value.
  - { "middleName": null }