

Variablen bezeichnen Speicherplätze.

Variablen mit Werten von Zahlen (char, short, int, long, long long, float, double, ...) bzw. Zeichen (char, ...) z.B. $\boxed{12}_x$ int x=12;

Variablen, die die Adressen von Variablen beinhalten, diese werden wie folgt dargestellt: (Adreßvariablen, Zeigervariablen)

$P_x = \boxed{\bullet} \rightarrow \boxed{12}_x$ oder $\boxed{\emptyset}$
int *px = &x; int x = 12; int *px = 0
zeigt auf nichts, Adresse 0

Zeigervariablen können auf Variablen im Stack verweisen (automatische Speicherplatzvergabe) (zur Übersetzungszeit)

oder
können auf dynamisch (mit malloc (C) bzw. new (C++, C#, Java, ...)) vergebenen Speicher (zur Laufzeit) verweisen:

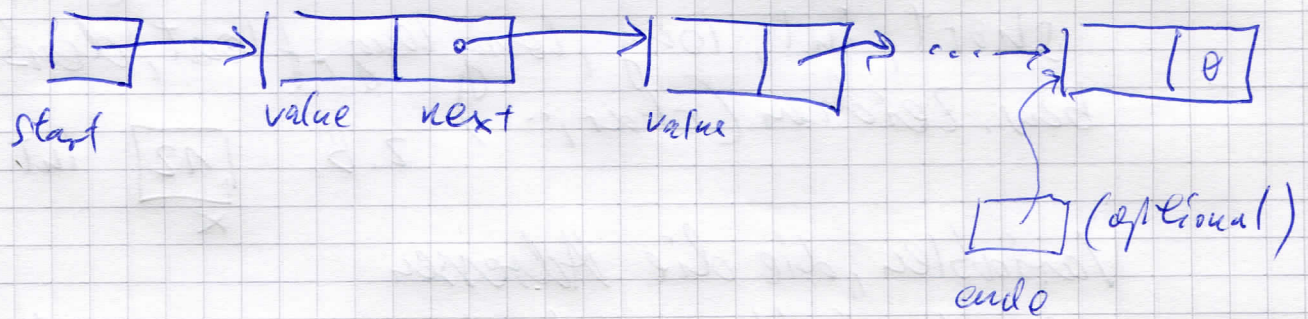
*/*C:*/* int *py = (int *) malloc(sizeof(int)); *py = 9;

*/*C++*/* int *py = new int(9); // *py = 12;

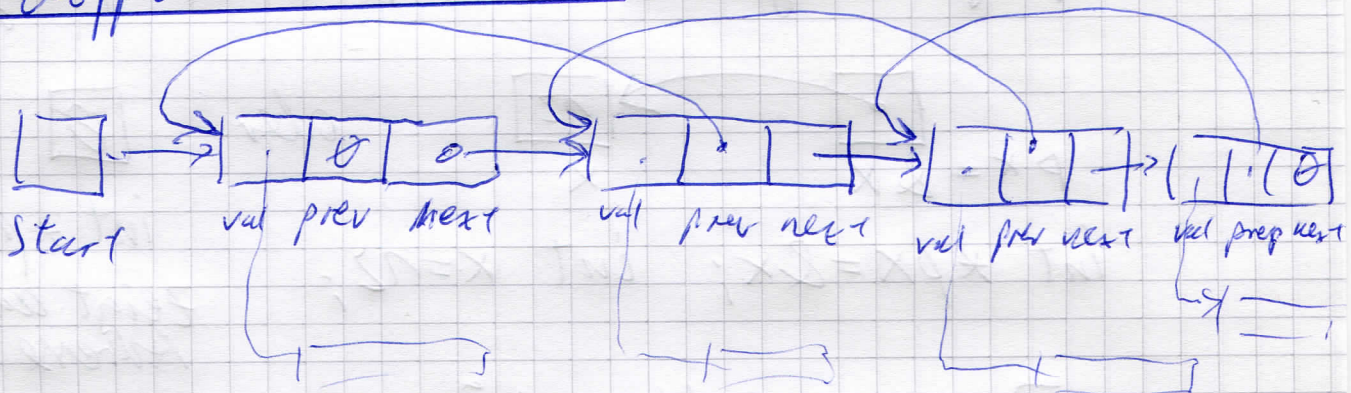
*/*Freigabe in C*/* free(py); py = 0;

*/*Freigabe in C++*/* delete py; py = 0;
bzw. py = nullptr;

Einfach verkettete Liste



Doppelt verkettete Liste



Implementierung in C

/ einfach verkettete Liste */*

```
typedef struct list {  
    int value; /* auch andere  
               Datentypen mögl. */  
    struct list *next; /* Zeiger */  
} list; /* neuer Typ list */
```

/ doppelt verkettete Liste */*

```
typedef struct dlist {  
    int value; /* andere Datentypen  
               mögl. */  
    struct list *prev; /* Zeiger */  
    struct list *next; /* Zeiger */  
} list2; /* neuer Typ list2 */
```