Tutorium 3

Stefan Heimersheim
stefan.heimersheim@rwth-aachen.de

13. November 2015

- Mehrere Variablen eines Typs
- Sequenz von Daten
- Konstante Größe
- Intern konstante Zeigern (beschränkte Zeigerarithmetik)

- `int i[3];`
- `const int n=3;`
- `int i[n];`
- `int i[3]={1,2,3};`
- `i[0]=1;`

- Mehrere Variablen eines Typs
- Sequenz von Daten
- Konstante Größe
- Intern konstante Zeigern (beschränkte Zeigerarithmetik)

- `int i[3];`
- `const int n=3;`
- `int i[n];`
- `int i[3]={1,2,3};`
- `i[0]=1;`

Hautpspeichermodell

- Addition einer ganzen Zahl k auf einen Zeiger p (Zeiger auf eine Variable des Typs T) liefert einen Zeiger gleichen Typs mit einem um $k \cdot sizeof(T)$ (k mal die Größe des von Variablen vom Typ T belegten Speichers in Byte) höheren Wert
- Eventuell nützlich für Navigation in Feldern
- Unübersichtlicher Code, Zugriff auf unsinnige Bereiche

Beispiel

Abstimmung: Was ist erlaubt?

`int const * b=&a`

- `cout « (*b)++;`
- `cout « *(b++);`
- `cout « *b++;`

Zeigerarithmetik auf Feldern:

- `cout « *(i+1);` geht
- `i=i+1;` geht nur bei dynamischen Feldern

Statische Felder sind als konstante Zeiger realisiert

Abstimmung: Was ist erlaubt?

`int const * b=&a`

- `cout « (*b)++;`
- `cout « *(b++);`
- `cout « *b++;`

Zeigerarithmetik auf Feldern:

- `cout « *(i+1);` geht
- `i=i+1;` geht nur bei dynamischen Feldern

Statische Felder sind als konstante Zeiger realisiert

Felder im Heap

- Größe variabel
- eigene Speicherverwaltung
- delete[] nicht vergessen

```
int i=3;
float* fv=new float[i];
fv[0]=10;
delete[] fv;
```

Hautpspeichermodell

- Mehrdimensionale Felder
- Matritzen, Tensoren, ...

```
int a[2][3];
int b[2][3]={{1,2,3},{4,5,6}};
int c[2][3]={1,2,3,4,5,6};
int d[2][3]={{1,2},{3,4},{5,6}}; //Fehler
```

```
int a[2][3]={{1,2,3},{4,5,6}};
    cout << &a[0][0] << endl;
    cout << &a[0][1] << endl;
    cout << &a[0][2] << endl;
    cout << &a[1][0] << endl;
    cout << &a[1][1] << endl;
    cout << &a[1][2] << endl;
```

```
0x7fff41d43010
0x7fff41d43014
0x7fff41d43018
0x7fff41d4301c
0x7fff41d43020
0x7fff41d43024
```

Hautpspeichermodell

Beispielcode:

```
int** m = new int* [2];
m[0] = new int[3];
m[1] = new int[3];
...
delete[] m[0];
delete[] m[1];
delete[] m;
```

Oft in for-Schleifen genutzt:

```
int n=2,m=3;
int** matrix = new int*[n];
for(int i=0;i<n;i++) matrix[i]=new int[m];
...
for(int i=0;i<n;i++) delete[] matrix[i];
delete[] matrix;
```

Hauptspeichermodell