WBS TRAINING

Programmierung(1)

UNSER ♥ SCHLÄGT FÜR BILDUNG

Agenda

- **Arrays**
  - Motivation + Beispielaufgabe
  - Darstellung in PAP, Struktogramm, Pseudocode
    - Initialisierung/Zuweisung
    - Auslesen
    - Nutzung in Schleifen
  - Syntax in ANSI C
    - Deklaration
    - Definition
      - Typ
      - Länge
        - » konstant
        - » Festlegung zur Laufzeit
    - Initialisierung/Zuweisung + Auslesen
    - Nutzung in Schleifen
    - Überschreiten der reservierten Felder
- Ausführliches Training + Ergebnisbesprechung
- Fachpraktische Anwendungen

Arrays – Motivation

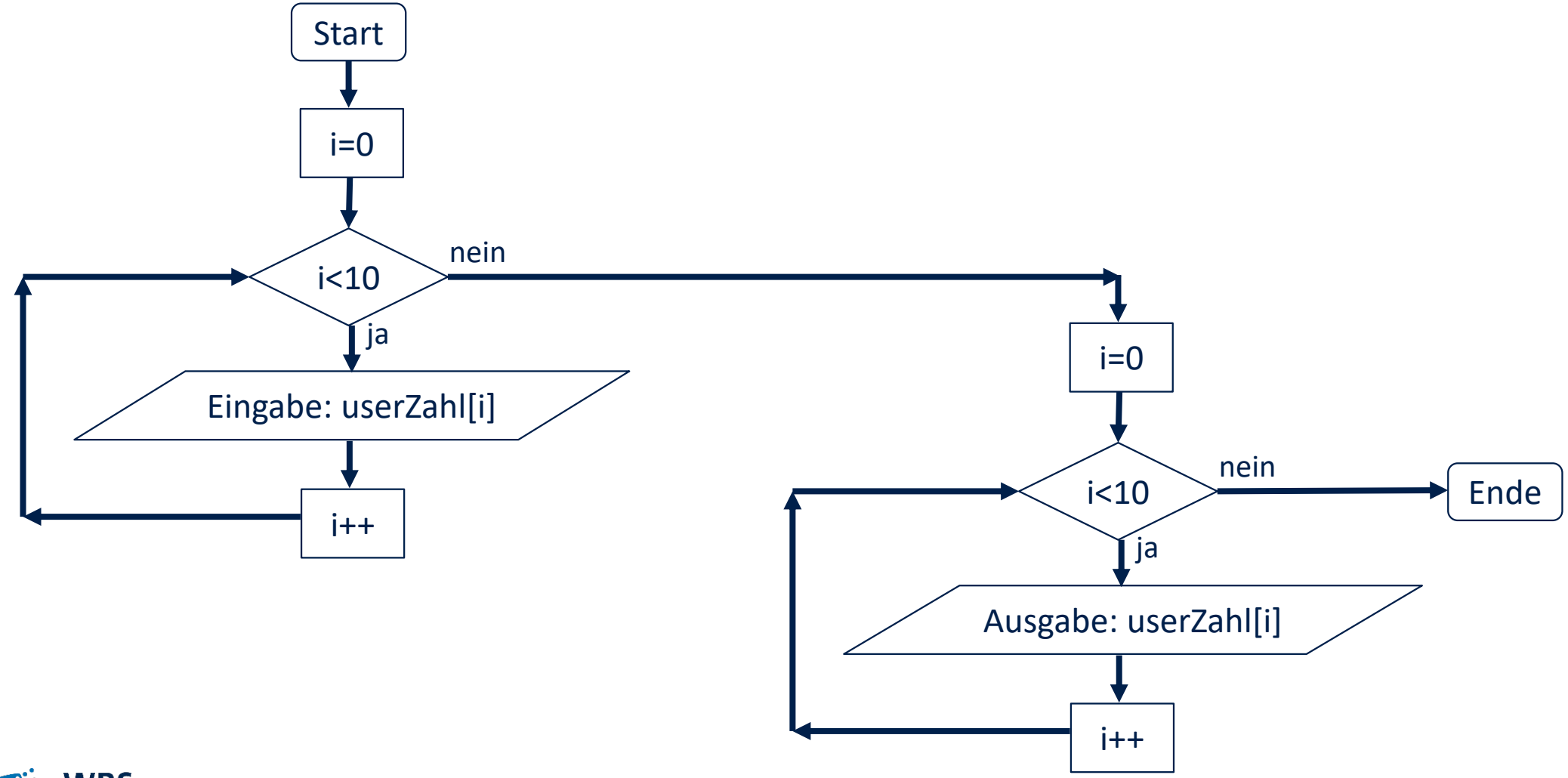
- Wir haben bereits ausführlich über Schleifen gesprochen, und uns dabei insbesondere auch mit den sogenannten „**Eingabe-Schleifen**“beschäftigt.
- Diese wurden von uns unter anderem aus den beiden folgenden Gründen verwendet:
  - Suche nach dem (z.B.) Maximum aller eingegebenen Zahlen
  - Erneute Eingabe-Aufforderung bei unzulässigen User-Eingaben
- Für all diese Eingabe-Schleifen galt bisher jedoch, dass die vom User gefüllte Variable in jedem einzelnen Durchlauf **überschrieben** wurde. Für das Programm war daher stets nur die jeweils letzte Eingabe bekannt. Alle zuvor getätigten Eingaben waren entsprechend verloren.
- Um dieses Problem zu lösen, nutzt man eine „Menge aus **durchnummerierten** Variablen“, die uns die folgende Vorgehensweise ermöglicht:
  - Im 1. Durchlauf wird vom User die 1. Variable der Menge gefüllt
  - Im 2. Durchlauf wird vom User die 2. Variable der Menge gefüllt
  - Im 3. Durchlauf wird vom User die 3. Variable der Menge gefüllt
  - … u.s.w. …
- Eine solche Menge „durchnummerierter Variablen“ wird als **Array** bezeichnet.
- Die einzelne Variable eines Arrays wird **Feld** genannt.
- Die konkrete Nummer eines einzelnen Feldes heißt **Index**.

Arrays – Beispielaufgabe

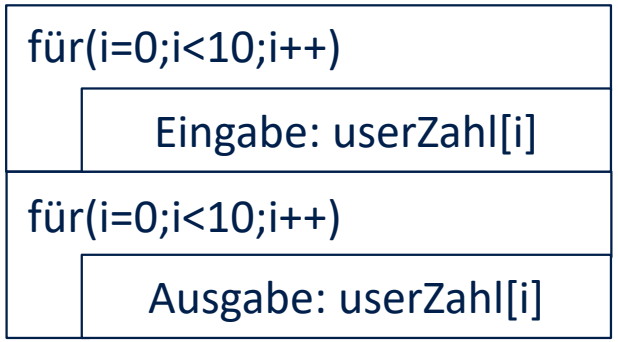
- Für Arrays gibt es kein eigenständiges Programmier-Symbol. Wir werden daher bei der Lösung der folgenden Beispielaufgabe (mittels PAP, Struktogramm oder Pseudocode) bereits jene Schreibweise kennenlernen, die im wesentlichen auch der Syntax des Quellcodes entspricht.
- Bei der Auswahl der Beispielaufgabe wurde darauf geachtet, dass wir alle wesentlichen Elemente im Umgang mit Arrays ansprechen. Dazu zählen neben der **Zuweisung** und dem **Auslesen** von Feldern auch die **Nutzung eines Arrays innerhalb von (Zähler)-Schleifen**.
- Im Rahmen der Codierung werden wir uns dann auch noch mit der **Deklaration** und **Definition** von Arrays zu beschäftigen haben.

Aufgabenstellung
Gegeben sei ein Array mit 10 Feldern. Mittels einer Eingabeschleife soll dem User die Möglichkeit gegeben werden, alle 10 Felder mit ganzen Zahlen zu füllen. Nach dieser Schleife soll zur Kontrolle der Inhalt aller 10 Felder auf der Konsole ausgegeben werden. Danach endet das Programm.

Beispielaufgabe – PAP



Beispielaufgabe – Struktogramm



Beispielaufgabe – Pseudocode

```
Programm „Array-Beispiel“
{
        für(i=0;i<10;i++)
        {
                Eingabe: userZahl[i]
        }
        für(i=0;i<10;i++)
        {
                Ausgabe: userZahl[i]
        }
}
```

Beispielaufgabe – Quellcode

```
# include<stdio.h>

main()
{
        int i;
        int userZahl[10];

        for(i=0;i<10;i++)
        {
                printf(„Geben Sie bitte eine ganze Zahl ein:“);
                fflush(stdin);
                scanf(„%d“,&userZahl[i]);
        }
        for(i=0;i<10;i++)
        {
                printf(„Wert des Feldes mit Index %d: %d\n“,i,userZahl[i]);
        }
}
```

Beispielaufgabe – Quellcode – Deklaration

```
# include<stdio.h>

main()
{
        int i;
        int userZahl[10];

        for(i=0;i<10;i++)
        {
                printf(„Geben Sie bitte eine ganze Zahl ein: );
                fflush(stdin);
                scanf(„%d“,&userZahl[i]);
        }
        for(i=0;i<10;i++)
        {
                printf(„Wert des Feldes mit Index %d: %d“,i,userZahl[i]);
        }
}
```

WBS TRAINING

Beispielaufgabe – Quellcode – Definition (Typ)

```
# include<stdio.h>

main()
{
        int i;
        int userZahl[10];

        for(i=0;i<10;i++)
        {
                printf(„Geben Sie bitte eine ganze Zahl ein: );
                fflush(stdin);
                scanf(„%d“,&userZahl[i]);
        }
        for(i=0;i<10;i++)
        {
                printf(„Wert des Feldes mit Index %d: %d“,i,userZahl[i]);
        }
}
```

Arrays können natürlich auch vom Typ **char**, **float**, oder **double** sein.
In jedem dieser Fälle gilt jedoch: **Alle Felder eines Arrays sind vom selben Typ!**

WBS TRAINING

Beispielaufgabe – Quellcode – Definition (konstante Länge)

```
# include<stdio.h>

main()
{
        int i;
        int userZahl[10];

        for(i=0;i<10;i++)
        {
                printf(„Geben Sie bitte eine ganze Zahl ein: );
                fflush(stdin);
                scanf(„%d“,&userZahl[i];
        }
        for(i=0;i<10;i++)
        {
                printf(„Wert des Feldes mit Index %d: %d“,i,userZahl[i]);
        }
}
```

Dieser Wert ist in sofern **„konstant“** als dass er bei jedem Programm-Aufruf stets den selben Wert (hier: 10) besitzen wird. Der Wert ist also schon vor der Kompilierung eindeutig festgelegt.

WBS TRAINING

Beispielaufgabe – Quellcode – Definition (variable Länge)

Dieser Wert ist in sofern **„variabel“** als dass er bei jedem Programm-Aufruf durch den User (oder einen Zufallsgenerator) neu bestimmt werden kann. Der Wert wird also erst zur Laufzeit eindeutig festgelegt.

```
…
…

main()
{
        …
        …
        zuf=rand()%1000;
        int userZahl[zuf];

        …
        …
        …
        …
        …
        …
        …
}
```

WBS TRAINING

Beispielaufgabe – Quellcode – Definition (variable Länge)

```
…
…

main()
{
        …
        …
        zuf=rand()%1000;
        int userZahl[zuf];

        …
        …
        …
        …
        …
        …
        …
}
```

„Intelligente“ Compiler kommen damit zu Recht. Da das kompilierte Programm allerdings nicht auf jeden System laufen wird, und da zudem die Bestimmung der Länge eines Arrays **zur Laufzeit des Programmes** nicht dem (aktuellen) ANSI C-Standard entspricht, wird im allgemeinen von einem solchen Vorgehen abgeraten.

WBS TRAINING

Beispielaufgabe – Quellcode – Überschreiten der Feldergrenzen

```
# include<stdio.h>

main()
{
        int userZahl[10];

        …
        …
        …
        …
        …
        …
        …
}
```

Beispielaufgabe – Quellcode – Überschreiten der Feldergrenzen

```
# include<stdio.h>

main()
{
        int userZahl[10];

        ...
        ...
        ...
        ...
        ...
        ...
        ...
}
```

Durch diese Deklaration und Definition wurde ein (Integer)-Array der **Länge 10** eingeführt. Für das Programm wurden somit die Felder userZahl[**0**], userZahl[**1**], userZahl[**2**], … bis userZahl[**9**] **reserviert**.

Beispielaufgabe – Quellcode – Überschreiten der Feldergrenzen

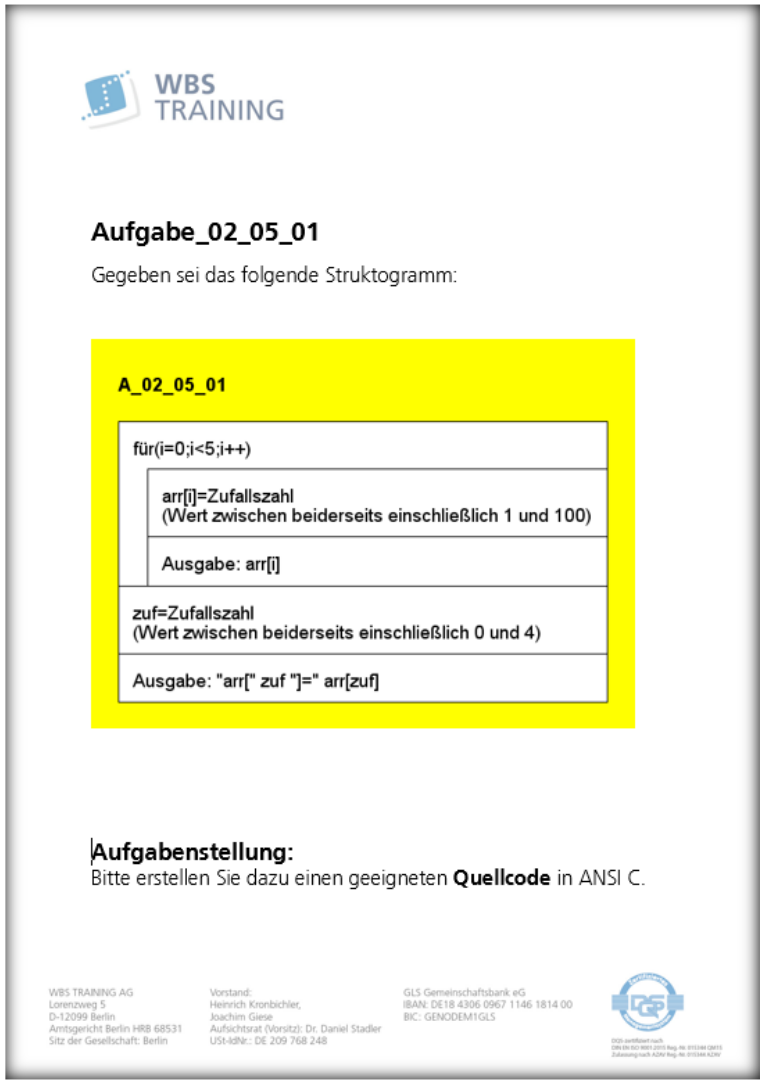
```
# include<stdio.h>

main()
{
        int userZahl[10];

        userZahl[-3]=77;
        userZahl[10]=1000;
        userZahl[15]=25;
        …
        …
        …
        …
}
```

Trotz dieser Deklaration und Definition kann man dann allerdings „gemeinerweise“ dennoch mit Feldern arbeiten, deren Speicherstellen vor, oder hinter dem reservierten Bereich liegen, OHNE dass dies vom Compiler als Fehler erkannt wird.

Da dies dazu führen kann, dass Ihr Programm gelegentlich einwandfrei läuft, bei anderen Durchläufen hingegen abstürzt, sind solche Fehler nicht immer leicht zu finden.

Der Programmierer sollte sich daher dieser Fehlerquelle bewusst sein, und sich bemühen, diese zu vermeiden!

Arrays – Gemeinsame Übung A_02_05_01



WBS TRAINING

Aufgabe_02_05_01

Gegeben sei das folgende Struktogramm:

A_02_05_01

- für(i=0;i<5;i++)
  - arr[i]=Zufallszahl (Wert zwischen beiderseits einschließlich 1 und 100)
  - Ausgabe: arr[i]
- zuf=Zufallszahl (Wert zwischen beiderseits einschließlich 0 und 4)
- Ausgabe: "arr[" zuf "]=" arr[zuf]

Aufgabenstellung:
Bitte erstellen Sie dazu einen geeigneten **Quellcode** in ANSI C.

WBS TRAINING AG
Lorenzweg 5
D-12099 Berlin
Amtsgericht Berlin HRB 68531
Sitz der Gesellschaft: Berlin

Vorstand:
Heinrich Kronbichler,
Joachim Giese
Aufsichtsrat (Vorsitz): Dr. Daniel Stadler
USt-IdNr.: DE 209 768 248

GLS Gemeinschaftsbank eG
IBAN: DE18 4306 0967 1146 1814 00
BIC: GENODEM1GLS

WBS TRAINING

VIELEN DANK FÜR IHRE AUFMERKSAMKEIT!