

# Übungsblatt Programmierertechnik IKA 9/13 – 04.06.2014

Lösen Sie alle Aufgaben in einer einzigen Projektdatei.

## Aufgabe 1:

Deklariieren Sie 6 einfache (elementare) Variablen vom Typ bool, int, char, float, double und string und initialisieren sie mit passenden Werten.

1. Lassen Sie sich anzeigen, wo im RAM diese Daten gespeichert werden.
2. Verändern Sie die Werte an diesen Adressen mit Hilfe eines Zeigers. Verwenden Sie für jedes Datum (jede zuvor deklarierte Variable) einen eigenen Zeiger.
3. Zeigen Sie die veränderten Werte an diesen Adressen an, indem Sie nun wieder die Variablennamen verwenden.
4. Erhöhen Sie jeden Zeiger um 1. Lassen Sie sich zu jedem Zeiger:
  - die neue Adresse anzeigen,
  - den Wert anzeigen, auf den dieser Zeiger nun zeigt. Was stellen Sie fest?

## Aufgabe 2:

Deklariieren Sie 6 Datenfelder vom Typ bool, int, char, float, double und string mit jeweils 5 Elementen und initialisieren sie mit passenden Werten.

1. Lassen Sie sich anzeigen, wo im RAM diese Datenfelder gespeichert werden.
2. Verändern Sie den Wert des jeweils dritten Elements der Datenfelder mit Hilfe eines Zeigers. Verwenden Sie für jedes Datenfeld einen eigenen Zeiger.
3. Zeigen Sie die veränderten Werte an diesen Adressen an, indem Sie nun wieder die Datenfeldnamen mit Index verwenden.
4. Geben Sie am Bildschirm
  - von den float- und den int-Datenfeldern
  - die Adressen jedes einzelnen Datenelements dieses Datenfelds sowie
  - den Wert der 5 Datenelementeaus. Verwenden Sie dafür jeweils eine for-Schleife.

## Aufgabe 3:

Hinweis:

Sie können diese Aufgabe mit Zeigern oder mit indexierten Datenfeldern lösen. Probieren Sie aber ruhig einmal beide Verfahren aus. Welche erscheint Ihnen komfortabler? Oder macht das keinen Unterschied?

1. Vertauschen Sie das jeweils letzte Element der in Aufgabe 2 erzeugten Datenfelder mit der einfachen Variablen aus Aufgabe 1.
2. Lassen Sie sich zur Kontrolle die Werte der 6 einfachen Variablen und die Werte der letzten Datenfeld-Elemente anzeigen – einmal vor der Vertauschung und einmal danach.

Speichern Sie Ihr Projekt auf USB-Stick, wir werden es in einem nächsten Praktikum (nach den Ferien, nach der Klausur) noch weiter entwickeln...

## Aufgabe 4 (Funktion mit Standardwerten):

Schreiben Sie eine Funktion **fahrtdauer**, die aus **Fahrtstrecke** (**dist** als Fahrtstrecke in km) und **Durchschnittsgeschwindigkeit** (**v** in km/h) die für eine Strecke benötigte Fahrtdauer berechnet und zurückgibt. Wird die Funktion ohne den Parameter **v** aufgerufen, so soll eine Standard-Durchschnittsgeschwindigkeit von 129 km/h angenommen werden.

### Bei genügend Zeit:

Schreiben und verwenden Sie eine zusätzliche Funktion, die das errechnete Ergebnis statt in Dezimalschreibweise schön in Stunden und Minuten (hh:mm) anzeigt.

## Aufgabe 5: Lokale Variablen vertauschen (Call by reference)

Definieren Sie IM HAUPTPROGRAMM 3 Double-Variablen und initialisieren sie mit beliebigen passenden Werten. Vermeiden Sie eine globale Deklaration!!!

Schreiben Sie eine Prozedur, die diese drei Werte zyklisch vertauscht:

- Die zweite Variable erhält den Wert der ersten Variablen,
- die dritte Variable erhält den Wert der zweiten Variablen und
- die erste Variable erhält den Wert der dritten Variablen.

Rufen Sie im Hauptprogramm die Prozedur auf und geben zur Kontrolle alle Variablenwerte VOR und NACH dem Tausch am Bildschirm aus.

## Aufgabe 6a: Call by reference (einfache Variable)

Schreiben Sie eine Funktion **quadrat**, die eine Ganzzahl als Referenz entgegennimmt und deren Wert quadriert. Die Funktion selbst gibt nichts aus und nichts zurück...

Definieren Sie im Hauptprogramm eine Ganzzahl (natürlich nicht gerade mit dem Wert 1!), rufen Sie die **quadrat**-Funktion 10 mal auf (Schleife!) und geben jedes Mal den aktuellen Wert der Ganzzahl aus.

## Aufgabe 6b: Call by reference (Array)

Jetzt das Gleiche mit Arrays:

Definieren Sie ein Array für 5 Ganzzahlen und initialisieren es mit geeigneten Werten.

Schreiben Sie eine Funktion **ArrayQuadrat**, die ein Array aus Ganzzahlen entgegennimmt und sämtliche Werte des Arrays quadriert...

Rufen Sie ebenfalls im Hauptprogramm 10 mal die Funktion **ArrayQuadrat** auf und lassen Sie sich alle Werte im Array anzeigen.

### Hinweis:

Hier sind mehrere Lösungen denkbar, zum Beispiel für die Anzeige eines Array-Inhalts eine weitere Funktion schreiben. Oder die Funktion **quadrat** aus Aufgabe 6a hier recyceln und in Ihrer neuen Funktion **ArrayQuadrat** aufrufen...