

*Programs must be written for people to read, and
only incidentally for machines to execute.*

Hal Abelson,

Gründungsmitglied der Free Software Foundation



UNIVERSITÄT
DES
SAARLANDES

FR Mathematik
Andreas Buchheit

2. Übung zur Vorlesung Programmierung im Sommersemester 2019

Abgabe: Mittwoch, den 24.04.2019 bis spätestens 12 Uhr.

Aufgabe 2.1. (6 Punkte) Datentypen

In dieser Aufgabe sollen Sie ihre Kenntnisse über C-Datentypen anwenden. Inkludieren Sie zunächst die auf der Vorlesungsseite verfügbare Header Datei 'employee.h'. Durch das inkludieren der Header Datei (Infos hierzu am Dateianfang) wird ein neuer Datentyp namens 'employee' definiert, der für die Speicherung der Daten von Angestellten eines (fiktiven) Unternehmens verwendet wird. Des Weiteren werden in der Header Datei drei Variablen des neuen Typs employee namens "alice", "bob" und "eve" definiert.

Kenntnisse über die Implementierung dieses Datentyps sind für die Lösung nicht erforderlich. Der Datentyp employee speichert grundlegende Datentypen in Felder mit den Bezeichnungen "name" (char []), "prename" (char []), "title" (char []), "position" (char []), "age" (int), "salary" (double), "bonus" (double), "section" (char) und "team_size" (size_t), wobei in Klammern der Datentyp angegeben ist. Die (char []) Einträge sind Null-terminierte Strings. Zugriff auf die Werte der einzelnen Felder erfolgt mit dem '.' Operator, das Alter von Eve erhält man z.B. mittels eve.age

- Geben Sie mithilfe von printf und den für den jeweiligen Datentyp passenden Platzhalter eine verständlich formatierte und vollständige Liste der Daten aller drei Angestellten aus. Verwenden Sie Tabs zur besseren Anordnung und geben Sie die Fließkommazahlen mit 2 Nachkommastellen aus.
- In dieser Teilaufgabe lernen Sie den Umgang mit Zufallszahlen kennen. Sehen Sie sich zunächst das Zusatzmaterial "random_number.pdf" an, in dem die Erzeugung einer Zufallszahl erklärt wird. Erweitern Sie dann ihr Programm aus (a), indem Sie vor der Ausgabe für alle drei Mitarbeiter dem Arbeitsbereich einen zufälligen Buchstaben von 'A' bis 'Z', der Teamgröße einen zufälligen Wert im Bereich von TEAM.SIZE_MIN bis TEAM.SIZE_MAX, und dem Bonus einen Wert von BONUS_MIN bis BONUS_MAX zuweisen, wobei die entsprechenden Makros in der header Datei definiert werden. Verwenden Sie type casts!

Aufgabe 2.2. (6 Punkte) Overflow

In dieser Aufgabe sollen der Speicherverbrauch unterschiedlicher Datentypen und der daraus folgende maximale Wertebereich untersucht werden. Inkludieren Sie zunächst die header "limits.h" und "float.h".

- Bestimmen Sie mithilfe des sizeof Operators den Speicherbedarf für Variablen des Typs char (signed und unsigned), int (signed und unsigned), size_t und double. Lesen Sie die Dokumentation von "limits.h" und "float.h". Diese Definitionsdateien definieren den Wertumfang von ganzen Zahlen (limits.h) und Fließkommazahlen (float.h). Geben Sie dann mit den in diesen Dateien definierten Makros den größten und kleinsten Wert für die oben genannten Datentypen sowie den Speicherbedarf in Byte aus.
- Deklarieren Sie eine int Variable und berechnen Sie durch Addition den maximalen Wert, bei der für eine int Variable overflow auftritt (while Schleife!). Geben Sie einen möglichst einfachen Weg an, um den maximalen Wert von size_t ohne limits.h zu auszugeben.