

Praxis der Programmierung

2. Aufgabenblatt

1. Schreiben Sie ein C-Programm wie folgt.
 - Deklarieren Sie eine globale Variable `y` vom Datentyp `int`.
 - Definieren Sie eine Funktion `f`: $\text{int} \times \text{int} \rightarrow \text{int}$, die den Quotienten aus dem Doppelten des Quadrats ihres ersten Parameters und dem Wert des zweiten Parameters berechnet.
 - Definieren Sie eine lokale Variable `x` der `main`-Funktion mit dem Datentyp `int` und dem Initialwert 4.
 - Weiterhin fordert die `main`-Funktion den Benutzer auf, eine ganze Zahl einzugeben und speichert dann diesen Wert als Initialwert von `y`. Danach wird die Funktion `f` mit `x` und `y` als aktuellen Parametern aufgerufen und der Rückgabewert auf die Konsole ausgegeben.
 - Testen Sie Ihr Programm mit verschiedenen Werten, die Sie für `y` eingeben. Was passiert, wenn Sie den Wert 0 eingeben?

2. Ergänzen Sie das Programm aus Aufgabe 1 so, dass die Funktion `f` für den Fall eines Aufrufs mit 0 als zweitem aktuellen Parameter eine geeignete Fehlerbehandlung durchführt.
3. Schreiben Sie ein C-Programm `pt1.c`, das die folgenden Definitionen von Variablen und die geforderten Anweisungen enthält:
 - Definition einer Variablen `i` vom Typ `int`,
 - Definition eines Pointers `ptr` auf `int`,
 - Zuweisung der Adresse von `i` an `ptr`,
 - Zuweisung des Wertes 1 an `i`,
 - Ausgabe des Wertes von `ptr`,
 - Ausgabe des Wertes des Speicherobjekts, auf das `ptr` zeigt,
 - Ausgabe des Wertes von `i`,
 - Zuweisung von 2 an das Speicherobjekt, auf das `ptr` zeigt (ohne Verwendung des Namens `i`,
 - Ausgabe des Wertes von `i`.

Hinweis: Pointerwerte gibt man bei `printf()` mit dem Formatelement `%p` aus.

4. Schreiben Sie ein C-Programm `swap.c`, das eine Funktion `swap()` mit zwei formalen Parametern enthält, die keinen Wert an den Aufrufer zurückgibt, sondern lediglich die Werte zweier `int`-Variablen gegeneinander austauscht.

Die `main`-Funktion definiert und initialisiert (mit verschiedenen Werten) zwei `int`-Variablen, die zunächst auf `stdout` ausgegeben werden, danach durch Aufruf von `swap()` vertauscht werden, und nach diesem Tausch noch einmal ausgegeben werden.