

Programmieren für Wirtschaftswissenschaftler SS 2015

Tutorium 2 – Aufgabenblatt

Inhalte

- Variablen – Deklaration, Initialisierung, Zuweisung
- Operatoren
- Eingabe, Verarbeitung und Ausgabe

Hinweis

Die Lösungen zu Präsenzübungen werden nicht veröffentlicht, sondern in der Übung besprochen.

Aufgabe 1 - Variablennamen

Entscheiden Sie, welche der nachfolgenden Variablennamen gültig sind. Falls es sich um keine gültige Variablen-Bezeichnung handelt, begründen Sie dies!

- 1) 3Personen
- 2) Verschachtelt=9
- 3) _dvd4-Titel
- 4) result
- 5) Resült
- 6) Result
- 7) R.e.s.u.l.t.

Aufgabe 2 – Variablen-Deklaration und -Initialisierung

Erläutern Sie anhand eines Beispiels, was eine Variablen-Deklaration und was eine -Initialisierung ist! Wie lässt sich beides miteinander verbinden?

Aufgabe 3 – Verschachtelte Ausdrücke

Es seien folgende Variablenbelegungen gegeben:

```
a = 5; b = 6; c = 7;
```

Verschachteln Sie den folgenden Ausdruck zu einem einzigen Ausdruck. Bedienen Sie sich der Assoziativität und Priorität der Operatoren und nutzen Sie dabei nur Klammern, sofern es nicht vermeidbar ist! Die Variablen a, b und c müssen nach der Ausführung des Ausdrucks dieselben Werte haben, wie dies nach den drei separaten Anweisungen der Fall wäre!

```
b = b + 1;  
c = a * b;  
a = a + 1;
```

Aufgabe 4 – Verschachtelte Ausdrücke

Setzen Sie statt der Variablen a, b und c (bzw. a++, ++a, etc.) die jeweilig genutzten Werte an den jeweiligen Positionen ein! Welchen Wert besitzt y nach der Ausführung?

```
int a = 9, b = 9, c = 9, y;  
y = a * ++b * c + c++ + ++c * c++;
```

Aufgabe 6 – Numerischer Überlauf

Kompilieren Sie die folgenden Programmzeilen und diskutieren Sie die Ausgabe!

```
byte b = byte.MaxValue-1;  
Console.WriteLine(++b);  
Console.WriteLine(++b);  
Console.WriteLine(++b);  
Console.WriteLine(++b);
```

Aufgabe 5 – Objektorientierung

Nachdem Sie schon wieder kein Geld mehr in der Tasche haben und echte Arbeit auch keine ernstzunehmende Alternative darstellt, entschließen Sie sich zur Gründung eines Mathematischen Beratungsinstituts. Als erste Dienstleistung möchten Sie Kreisberechnungen anbieten. Dazu haben sie den folgenden Code erstellt.

- Welche Elemente enthält die Klasse Kreis?
- Beschreiben Sie den Ablauf des Programms!
- Erweitern Sie den Code um eine Funktion, mit der die Kreisfläche berechnet werden kann!
- Erweitert:* Wie können Sie erreichen, dass nur zwei Nachkommastellen der berechneten Werte in der Konsole angezeigt werden?

```
class Program  
{  
    static void Main(string[] args)  
    {  
        Kreis k1 = new Kreis(20); //erstelle Objekt k1  
        Console.WriteLine(k1.BerechneUmfang()); //berechne Umfang von k1 und gib diesen aus  
        Kreis k2 = new Kreis(30); //erstelle Objekt k2  
        Console.WriteLine(k2.BerechneUmfang()); //berechne Umfang von k1 und gib diesen aus  
        Console.ReadLine();  
    }  
}  
  
class Kreis  
{  
    double radius;  
  
    //Konstruktor  
    public Kreis(double r)  
    {  
        radius = r; //Wert von r in radius speichern  
    }  
  
    //Funktion zur Berechnung des Umfangs  
    public double BerechneUmfang()  
    {  
        return 2 * Math.PI * radius * radius;  
    }  
}
```

[Zusatzaufgabe 7 – Taschenrechner mit Benutzereingaben]

Programmieren Sie einen einfachen Taschenrechner! Der Benutzer gibt dazu zwei Zahlen ein, zu dem das Programm dann folgende Berechnungen durchführt und die jeweiligen Ergebnisse ausgibt:

$$x + y, \quad x - y, \quad x \cdot y, \quad \frac{x}{y}, \quad e^x, \quad \min(x, y), \quad \max(x, y)$$

Nutzen Sie als Grundlage den am Ende dieser Aufgabe stehenden Code als Startpunkt. Machen Sie sich bewusst, was die dortigen Anweisungen bewirken.

Information zu den mathematischen Funktionen finden Sie auf der folgenden Seite des MSDN:

<http://msdn.microsoft.com/en-us/library/system.math.aspx>

Ihr Programm speichert die beiden eingegebenen Zahlen als Integer-Variablen (x und y). Konvertieren Sie die Variablen bei Bedarf in einen anderen Datentyp, um z.B. Nachkommastellen anzuzeigen. Zum Ende werden die Ergebnisse der einzelnen Terme ausgegeben.

```
static void Main(string[] args)
{
    string input1 = Console.ReadLine();
    int zahl1 = Convert.ToInt32(input1);

    string input2 = Console.ReadLine();
    int zahl2 = Convert.ToInt32(input2);

    //Berechnungen...

    Console.ReadLine();
}
```

Erweitert: Wie können Sie sicherstellen, dass die eingelesenen Werte für zahl1 und zahl2 gültige Zahlen darstellen?