



# Základy programování a algoritmizace Aritmetické výrazy

# Erik Král



2020

#### Informace o autorech:

Ing. et Ing. Erik Král, Ph.D.
Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Nad Stráněmi 4511
760 05 Zlín
ekral@utb.cz



## **OBSAH**

OBSAH			:
		OD	
		ARITMETICKÉ VÝRAZY	
		Řešené příklady	
SEZ	ZNAM	POUŽITÉ LITERATURY	13





## 1 ÚVOD

V tomto materiálu se seznámíme s aritmetickýmy výrazy a projdeme si i příklady na procvičení.

#### 1.1 Aritmetické výrazy

Výrazy se skládají z operandů a operátorů. Operátory ve výrazu určují, která operace se aplikuje na operandy. Operandy mohou být například proměnné nebo konstanty.

V následujících příkladech probereme aritmetické operátory. Nejprve si definujeme tři proměnné x, y a z.

```
double x = 2.0;
double y = 3.0;
double z = 0.0;
```

Proměnné x a y potom použijeme jako operandy binárního operátoru sčítání. Jak už z názvu vyplývá, binární operátor je takový operátor, který vyžaduje dva operandy.

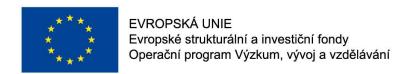
```
z = x + y;
```

Dalšími binární aritmetickými operátory jsou rozdíl, součin a podíl:

```
z = x - y; // rozdil
z = x * y; // soucin
z = x / y; // podil
```

Kromě binárních operátorů máme i unární operátory, kdy operand je jen jeden. Příkladem je operátor vracející zápornou hodnotu:

```
z = -x;
```







Dalšími unárními operátory je operátor inkrementace a dekrementace, ten může být použití jako prefixový, tedy před názvem proměnné, tak postfixový, tedy za názvem proměnné. Prefixová varianta inkrementace a dekrementace vypadá následovně, kdy první příklad zvýší hodnotu proměnné o 1 a druhý příklad tuto hodnotu o 1 sníží.

```
++z;
--z;
```

Postfixová varianta vypadá následovně:

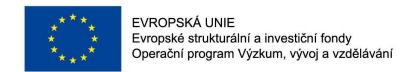
```
z++;
z--;
```

Rozdíl je v v pořadí vyhodnocení výrazu a provedení operace. V případě postfixové varianty nejprve získáme hodnotu a teprve potom se provede inkremetace. U postfixové varianty se nejprve hodnota zvýší a potom ji získáme. Například v následujícím kódu bude mít na konci programu proměnná y hodnotu 0 a proměnná x hodnotu 1.

```
static void Main(string[] args)
{
   int x = 0;
   int y = x++;

   Console.WriteLine($"{x} {y}");
}
```

V programech často využíváme matematické operace jako je například mocnina, odmocnina anebo konstanta  $\pi$ . Tyto operace a konstanty najdeme jako statické metody ve statické třídě Math.







Následující příklad spočítá mocninu  $x^{100}$  a druhou odmocninu  $\sqrt{x}$ . Tyto operace jsou nadefinovány pouze pro typ *double*.

```
using System;

double x = 2.0;
double z = 0.0;

z = Math.Pow(x, 100.0);
z = Math.Sqrt(9.0);
```

Pro druhou mocninu je ale vhodnější použí operátor součinu:

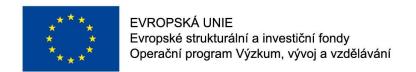
```
z = x * x;
```

Následující příklad potom vypíše na terminál hodnotu konstanty π:

```
Console.WriteLine(Math.PI);
```

Určování priorit operátorů můžeme pomocí kulatých závorek (). Prioritu operátorů najdete v dokumentaci jazyka C# [1].

```
z = x * (y + 3.0);
```







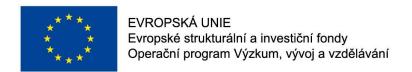
Dále můžeme použít zápis compound assignement (složené přířazení), kdy následující výrazy:

```
double a = 1.0;

a = a + 2.0;
a = a - 2.0;
a = a * 2.0;
a = a / 2.0;
```

můžeme zapsat zkráceně:

```
a = a + 2.0;
a = a - 2.0;
a = a * 2.0;
a = a / 2.0;
```







### 1.2 Řešené příklady

Nyní si vše probereme na příkladech. První příklad spočítá a vypíše na konzoli **obvod a obsah čtverce** definovaného délkou strany v proměnné *n*.

```
using System;

namespace MujPrvniProjekt
{
    class Program
    {
        static void Main(string[] args)
        {
            double n = 3.0;
            Console.WriteLine($"Delka strany je {n}");

            double obvod = 4 * n;
            Console.WriteLine($"Obvod je {obvod}");

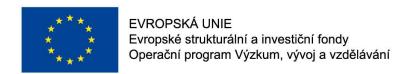
            double obsah = n * n;
            Console.WriteLine($"Obsah je {obsah}");
        }
    }
}
```

Obvod čtverce určíme jako čtyřnásobek délky strany:

```
double obvod = 4 * n;
```

Všimněte si, že pro výpočet druhé odmociny pro obsah čtverce byl použit operátor součinu:

```
double obsah = n * n;
```







Druhý příklad spočítá a vypíše **obvod a obsah kruhu** definovaného poloměrem v proměnné *r*. Příklad demonstruje použití konstanty *Math.PI*.

```
using System;

namespace MujPrvniProjekt
{
    class Program
    {
        static void Main(string[] args)
        {
            double r = 3.0;
            Console.WriteLine($"Polomer kruhu je {r}");
            Console.WriteLine($"Hodnota Pi je {Math.PI}");

            double obvod = 2 * Math.PI * r;
            Console.WriteLine($"Obvod je {obvod}");

            double obsah = Math.PI * r * r;
            Console.WriteLine($"Obsah je {obsah}");

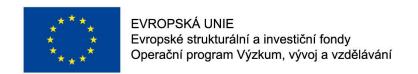
            }
        }
    }
}
```

Obvod kruhu je vypočítán pomocí tohoto příkazu:

```
double obvod = 2 * Math.PI * r;
```

A obsah kruhu je potom určen tímto příkazem. Opět si všimněte výpočtu druhé odmocniny r \* r:

```
double obsah = Math.PI * r * r;
```







Další příklad spočíta a vypíše hodnotu BMI (Index tělesné hmotnosti) [2] dle výšky v metrech a hmostnosti v kilogramech.

```
using System;

namespace MujPrvniProjekt
{
    class Program
    {
        static void Main(string[] args)
        {
            double hmotnost = 85;
            double vyska = 1.78;

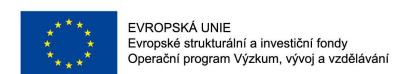
            Console.WriteLine($"hmotnost {hmotnost}kg a vyska {vyska}m");

            double bmi = hmotnost / (vyska * vyska);

            Console.WriteLine($"bmi je {bmi}");
            }
        }
}
```

Všimněte si využití kulatých závorek pro prioritu operátorů:

```
double bmi = hmotnost / (vyska * vyska);
```





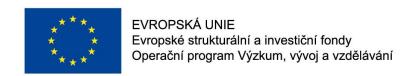


Další příklad určí obvod a obsah trojúhelníku zadaného délkami stran pomocí Heronova vzorce [3]. Opět si musíme dát pozor na prioritu operátorů a využijeme také metodu *Math.Sqrt* pro druhou odmocinu.

```
using System;
namespace MujPrvniProjekt
{
    class Program
        static void Main(string[] args)
            double a = 3.0;
            double b = 4.0;
            double c = 5.0;
            Console.WriteLine($"Delka strany trojuhelnika je {a}, {b} a {c}");
            double obvod = a + b + c;
            Console.WriteLine($"Obvod je {obvod}");
            double s = (a + b + c) / 2;
            double obsah = Math.Sqrt(s * (s - a) * (s - b) * (s - c));
            Console.WriteLine($"Obsah je {obsah}");
        }
    }
```

Při výpočtu obsahu program využívá metodu Math. Sqrt pro výpočet druhé odmocniny:

```
double obsah = Math.Sqrt(s * (s - a) * (s - b) * (s -c));
```







Poslední příklad demostruje využítí metody *Math.Pow* pro výpočet mocniny čísla. Konrétně spočítá a vypíše výši pravidelné splátky dle zadané výši dluhu, úrokové míře a délky splácení [4].

```
using System;

namespace ConsoleApp11
{
    class Program
    {
        static void Main(string[] args)
        {
            int pocetLetSplaceni = 20;
            double rocniUrokProcenta = 2;
            double D = 1000000; // dluh

            int n = pocetLetSplaceni * 12; // pocet mesicu splaceni
            double i = rocniUrokProcenta / (12 * 100); // desetinne cislo

            double v = 1 / (1 + i);
            double splatka = (i * D) / (1 - Math.Pow(v, n));

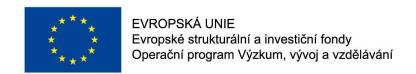
            Console.WriteLine($"Mesicni splatka bude {splatka:F2} Kc");
        }
    }
}
```

Na tomto řádku využívá program metodu Math. Pow:

```
double splatka = (i * D) / (1 - Math.Pow(v, n));
```

A na tomto řádku používáme zápis splatka: F2 který formátujeme výpis splátky na zobrazení dvou desetinných čísel.

```
Console.WriteLine($"Mesicni splatka bude {splatka:F2} Kc");
```







## SEZNAM POUŽITÉ LITERATURY

- [1] Precedence and order of evaluation | Microsoft Docs. [online]. Copyright © Microsoft 2020 [cit. 02.10.2020]. Dostupné z: https://docs.microsoft.com/en-us/cpp/c-language/precedence-and-order-of-evaluation?view=msvc-160
- [2] Index tělesné hmotnosti Wikipedie. [online]. Dostupné z: https://cs.wikipedia.org/wiki/Index\_t%C4%9Blesn%C3%A9\_hmotnosti
- [3] Obsah trojúhelníku Matematika.cz. Matematika pro střední a základní školy Matematika.cz [online]. Copyright © 2006 [cit. 02.10.2020]. Dostupné z: https://matematika.cz/obsah-trojuhelniku
- [4] Výpočet hypotéky [online] [cit. 02.10.2020]. Dostupné z: http://www.aristoteles.cz/matematika/financni\_matematika/hypoteka-vypocet.php

