

Třídy a objekty

4. cvičení

Jiří Zacpal

KMI/ZP3CS – Základy programování 3 (C#)

Definování třídy

- syntaxe:

```
class identifikátor {}
```

- příklad:

```
class Kruh
{
    int polomer; //datová složka

    double Plocha()
    {
        return
Math.Pi*polomer*polomer;
    }
}
```

Použití třídy

- vytvoření objektu = instance třídy

```
jmeno_tridy identifikator =new  
jmeno_tridy()
```

příklad:

```
Kruh k;
```

```
k=new Kruh();
```

Řízení přístupu

- `public` – veřejný přístup
- `private` – přístupné pouze uvnitř třídy
- příklad:

```
class Kruh
{
    private int polomer;
    public double Plocha()
    {
        return
Math.Pi*polomer*polomer;
    }
}
```

Konstruktor

- metoda, která se spustí při vytvoření třídy

```
public jmeno_tridy() { }
```

- příklad:

```
public Kruh()  
{  
    polomer=0;  
}
```

Přetěžování konstruktoru

- třída může obsahovat více konstruktoru, které se liší typem nebo počtem parametrů
- příklad

```
public Kruh()  
{  
    polomer=0;  
}  
public Kruh(int r)  
{  
    polomer=r;  
}
```

- použití

```
Kruh k1=new Kruh();  
Kruh k2=new Kruh(8);
```

Statické metody a data

- statické metody lze volat i bez existence objektu
- statická metoda má přístup pouze ke statickým datovým složkám
- příklad:

```
class Math()  
{  
    public static double Sqrt(double  
d) { }  
}
```

- použití:

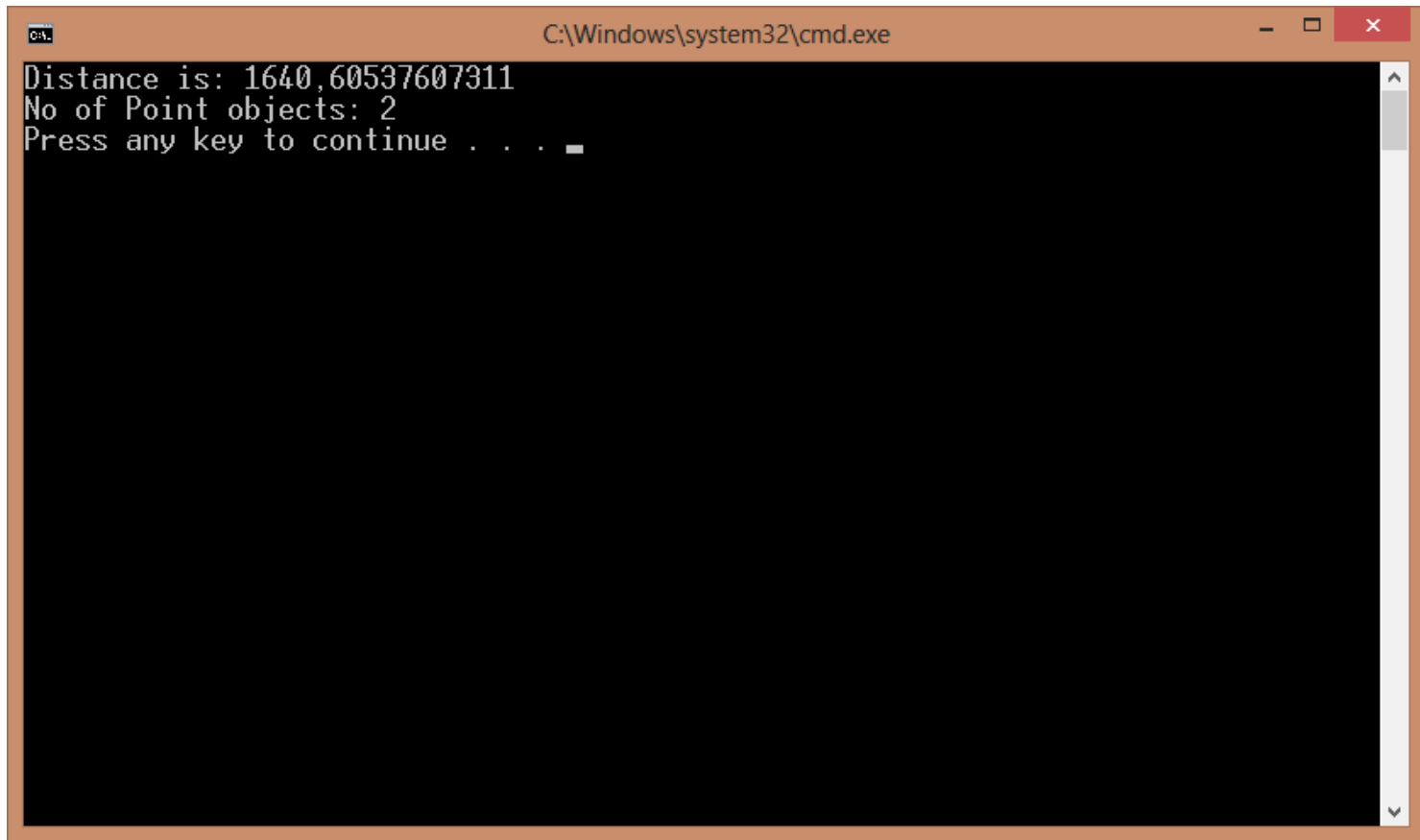
```
Math.Sqrt(10);  
Kruh k2=new Kruh(8);
```

Statické třídy

- třída, která obsahuje pouze statické metody a data
- nelze z ní vytvářet objekty
- příklad:

```
static class Math()  
{  
    public static double  
    Sqrt(double x) {}  
    public static double Sin(double  
x) {}  
}
```


Příklad 1



A screenshot of a Windows command prompt window. The title bar at the top reads "C:\Windows\system32\cmd.exe". The window has a black background with white text. The text displayed is:

```
Distance is: 1640,60537607311  
No of Point objects: 2  
Press any key to continue . . . █
```

The cursor is positioned at the end of the third line. A vertical scrollbar is visible on the right side of the window.

Kopírování proměnných a tříd

- hodnotový typ – int, float, double, char,...
- referenční typy – string, třídy

Hodnota null

- speciální hodnota, kterou lze přiřadit jakékoli referenční proměnné
- příklad:

```
Kruh k=null;
```

- nulovatelný typ:

```
int? i=null;
```

Předávání parametru

- hodnotou
- referencí - ref

```
int f(ref int p) { }
```

- výstupní parametry - out
 - na rozdíl od referencí nemusí být při předávání inicializovány
 - ve funkci jim musí být přiřazena hodnota

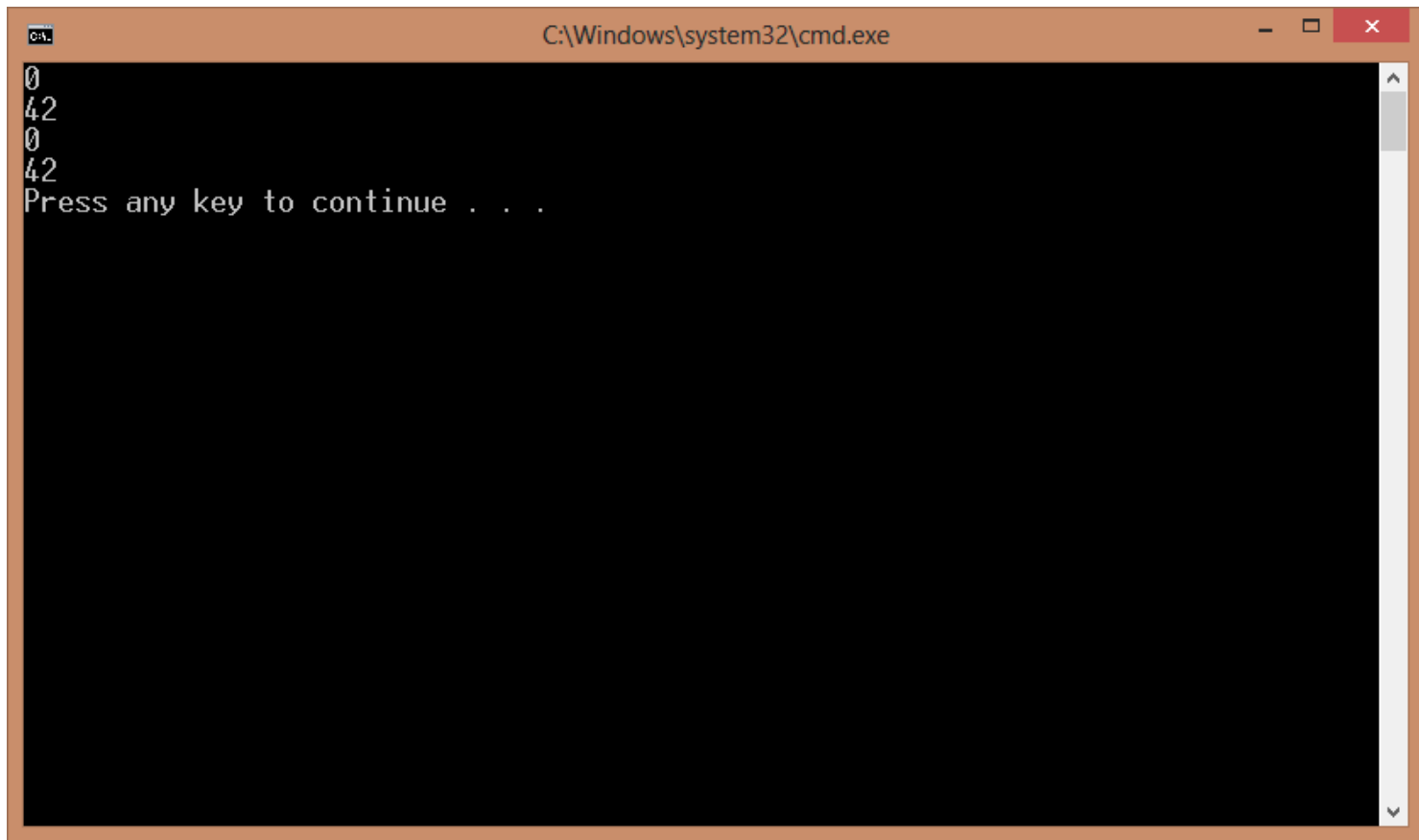
```
int f(out int p) { p=10; }
```

- volání

```
int x;
```

```
f(out x);
```

Příklad 2



A screenshot of a Windows command prompt window. The title bar at the top reads "C:\Windows\system32\cmd.exe". The command prompt shows the following output:

```
0  
42  
0  
42  
Press any key to continue . . .
```

The window has a standard Windows interface with a title bar, a maximize button, and a close button. A vertical scrollbar is visible on the right side of the command prompt area.

Destruktor

- metoda, která se vykoná při rušení objektu
- příklad:

```
class Citac
{
    public static int pocet=0;
    public Citac()
    {this.pocet++;}

    ~Citac()
    {this.pocet--;}
}
```

Omezení destruktorků

- destruktory nemohou být public
- destruktor nemůže mít parametry
- destruktor se vykoná při rušení objektu
 - nevíme kdy
 - automatické čištění paměti