

Jazyk C# a platforma .NET

Katedra softwarového inženýrství
Fakulta informačních technologií
České vysoké učení technické v Praze

© Pavel Štěpán, Helena Wallenfelsová 2014

Doplňky pro práci s pamětí
BI-DNP



```

// GC (Garbage Collector) automaticky uvolňuje (casem) pamet objektu,
// na které již nevede pointer z programu. Ale pro objekty, které používají
// unmanaged zdroje je třeba definovat (a vyvolávat před zrušením) metodu
// Dispose. Její automatické vyvolání zajišťuje blok using. Je to vhodný
// přístup, pokud chceme použít objekt v rámci jednoho bloku a po jeho
// objekt zrušit a „uklidit“ unmanaged zdroje

void UkazkaPouzitiUsing(){
    // po opuštění bloku using je automaticky volána metoda Dispose!!
    // (pro použití v using MUSÍ třída (Zam) implementovat IDisposable)
    // nepřesť s using na začátku programu!!

    using (Zam objZam = new Zam()){
        // ...
        objZam.ID = 1;
        // ...
    }
}

// Použití Weak References umožňuje GC zrušit při nedostatku paměti i objekty,
// na které odkazují z programu pointery, ovšem typu weak references.
// Vhodné pro paměťové náročné programy.
// Použita stejná třída Zam jako v předchozím příkladu, ale jen proto, abychom
// nemuseli definovat další třídu. Zde se nemusí implementovat IDisposable

void UkazkaPouzitiWeakReferences(){
    // GC nezruší objekt, pro který existuje pointer z programu.
    // Při nedostatku paměti by ale bylo vhodné zrušit
    // "obnovitelné" objekty (např. s daty načítanými z DB).
    // To umožní weak references:

    Zam objZam = new Zam();
    // vytvořím weak referenci na objekt
    WeakReference wr = new WeakReference(objZam);
    objZam = null; // zruším "tvrdou" referenci

    // ... během těchto operací mohl GC zrušit objekt s wr

    // další práce s objektem
    objZam = (Zam)wr.Target;
    if (objZam != null){
        // objekt nebyl zrušen - možno používat
    }
    else {
        // objekt neexistuje - pro použití nutno obnovit
    }
}

```

```
// Trida, pouzita v prikladech. Kvuli prvnimu prikladu implementuje
// interface IDisposable, který je definovan v .NET (jedina metoda Dispose)

class Zam : IDisposable {
    public int ID;
    public string Jmeno;

    // dalsi metody a vlastnosti (napr. i operace s DB, soubory, ...)

    public void Dispose(){
        // zde mozno provest "uklidove" operace
        // (zavrit connection k DB, zavrit soubory, ...)
    }
}
```