GRASP – Controller

- Součást MVC architektury (ovladač)
- Jeho role spočívá v komunikaci s uživatelem
- Logika je tím pádem zcela odstíněna od prezentace
- Na následujících stránkách si ukážeme špatný a dobrý příklad

GRASP – Controller (kalkulačka)

Porušení principu

- smíchání komunikace s logikou
- side effect – není univerzální + skrytá komunikace s konzolí

```
public int Secti()
{
        Console.WriteLine("Zadej 1. číslo");
        int a = int.Parse(Console.ReadLine());
        Console.WriteLine("Zadej 2. číslo");
        int b = int.Parse(Console.ReadLine());
        return a + b;
}
```

GRASP – Controller (kalkulačka)

Porušení principu 2

- zápis logiky do obslužných metod GUI
- obslužná třída (controller) je znečištěn logikou (výpočtem)

```
public void SectiTlacitko_Click(Object sender)
{
        int a = int.Parse(cislo1.Text);
        int b = int.Parse(cislo2.Text);
        vysledekLabel.Text = (a + b).ToString();
}
```

Ve všech aplikacích má být jedna vrstva sloužící pouze ke komunikaci

- 1. příklad – vrstva zcela chybí
- 2. příklad – dělá více věcí najednou

GRASP – Controller (kalkulačka)

Řešení 1 – konzolová kalkulačka

- main() – součást controlleru, pouze komunikace
- logika – třída Kalkulacka

```
public static function main()
{
        Kalkulacka kalkulacka = new Kalkulacka();
        Console.WriteLine("Zadej 1. číslo");
        int a = int.Parse(Console.ReadLine());
        Console.WriteLine("Zadej 2. číslo");
        int b = int.Parse(Console.ReadLine());
        Console.WriteLine(kalkulacka.Secti(a, b));
}
```

GRASP – Controller (kalkulačka)

Řešení 2 – třída pro Controller

- Controller – parsování vstupu, změna obsahu labelu
- Kalkulacka – pouze výpočet, neví nic o formuláři

```
class KalkulackaKontroler
{

        private Kalkulacka kalkulacka = new Kalkulacka();

        public void SectiTlacitko_Click(sender: Object)
        {
                int a = int.Parse(cislo1.Text);
                int b = int.Parse(cislo2.Text);
                vysledekLabel.Text = (kalkulacka.Secti(a, b)).ToString();
        }

}
```