

UPR

Cvičení 3

Pointer (ukazatel)

- Proměnná uchovávající adresu do paměti
 - Data nemusíme kopírovat, ale odkážeme se na ně

```
int *a; // ukazatel na integer
```

```
float *b; // ukazatel na float
```

- Do ukazatele se ukládá adresa (jak zjistit adresu proměnné?
Pomocí &)

```
int a = 5;
```

```
int *b = &a;
```

Pointer (ukazatel)

- Jak získat hodnotu pointeru?

```
printf("Adresa: %p, hodnota: %d\n", b, *b);
```

- K čemu je to dobré? Např. když chceme z funkce vrátit více hodnot.

Pointer (ukazatel)

- Úkol č.1
 - Vytvořte funkci pro součet dvou čísel
 - Upravte funkci, že vrátí součet i rozdíl těchto čísel
 - Buď vraťte jednu hodnotu pomocí `return` a druhou pomocí ukazatele nebo obě pomocí ukazatele (funkce pak bude vracet `void`)

Pointer (ukazatel)

- Úkol č.2

- Vytvořte funkci `quadr_roots`, která vrátí počet kořenů kvadratické rovnice $ax^2+bx+c=0$ pomocí `return` a vypočítané kořeny vrátí pomocí předaných ukazatelů v argumentech funkce.

```
int quadr_roots(float a, float b, float c, float *x1, float *x2);
```

- Počet kořenů lze zjistit vypočítáním diskriminantu $D=b^2-4ac$. Pokud vyjde diskriminant záporný, tak funkce vrátí nulu, protože žádné řešení v oboru reálných čísel neexistuje. Pro nulový diskriminant funkce vrátí 1 a uloží dvojnásobný kořen na adresu ukazatelů x_1, x_2 . Pro kladný diskriminant funkce vrátí 2 a vypočítá kořeny pomocí:

$$x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$$