

15 Pole I

Častý problém při programování: potřebujeme uložit více hodnot stejného typu. Například v programu pro výpočet mezd, potřebujeme 12 proměnných pro dvanáct měsíců v roce. Jediné co teď umíme:

```
long mesic1, mesic2, ... mesic12;
```

⇒ spousta nedostatků:

- strašná pracnost (u pracovníka zaměstnaného 10 let už jde o 120 proměnných)
- není možné s tím snesitelně pracovat:
příkaz pro výpočet roční mzdy $rocni = mesic1 + mesic2 + \dots + mesic12$;
pro dalšího zaměstnance musím napsat jiný kód

⇒ takto se s větším množstvím dat pracovat nedá

**Řešení = pole (možnost jak vytvořit více proměnných stejného typu „na jednou“)
deklarace:**

```
long mesic[12]      - počítač vytvoří v paměti za sebou 12 proměnných typu long,  
jednotlivé proměnné mají označení mesic[0], mesic[1], ..., mesic[11]
```

výhoda polí není v rychlosti deklarace, ale ve způsobu zpracování:

sečtení mezd v jednom roce pomocí pole

```
for(i=0; i<12;i++) rocni+=mesic[i];
```

když budu potřebovat sečíst 120 položek stačí změnit jen podmínku v cyklu

```
for(i=0; i<120;i++) rocni+=mesic[i];
```

⇒ v tomto okamžiku konečně začíná programování

Přečti úvod článku (do ukázky třídění) o polích na adrese

http://www.linuxsoft.cz/article.php?id_article=560

Př. 1: Napiš program, do kterého uživatel zadá deset celých čísel, program je uloží do pole a poté nabídne uživateli možnost vypsání libovolného čísla v poli (třeba čísla zadaného jako třetí). Nabídku bude opakovat, dokud uživatel nezadá číslo, které program nemůže mít uložené (třeba 12.)

Př. 2: Předchozí program uprav tak, aby si uživatel mohl zvolit zda program vypíše čísla: popořadě, v obráceném pořadí nebo nejdříve popořadě čísla na sudých místech pak čísla v obráceném pořadí čísla na lichých místech.

Př. 3: Předělej program tak, aby umožňoval zjistit, jestli se v poli nachází určitá hodnota (třeba číslo 11) pokud ho najde vypíše program pozici, na které je v poli uloženo.

|| **Př. 4:** Aasd