

Základy procedurálneho programovania 1

Prednáška č. 1

Čím žijú programy?

19. 9. 2016

zimný semester
2016/2017

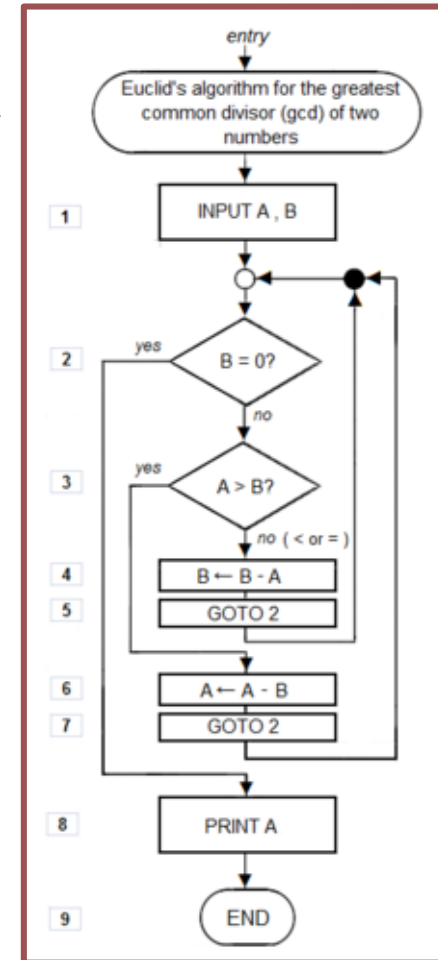
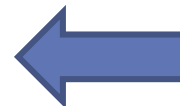
I Programovanie vs. Programovací jazyk

- **Počítač** je nástroj
- **Program** je postupnosť inštrukcií, ktoré povedia počítaču, ako vykonať úlohu
- Počítač **vykonáva** program
- Vykonáva **presne** čo je v programe
- Inštrukcie procesora sú zapísané binárne (0/1)
 - Náročné a zdĺhavé písať takéto programy
 - Pre sprostredkovanie inštrukcií počítaču používame **programovacie jazyky**
- **Programovanie** je vymyslenie a navrhnutie postupu riešenia úlohy a zapísanie tohto postupu v programovacím jazyku

I Úloha → algoritmus → program

**Dané sú čísla A a B,
vypočítaj ich najväčší
spoločný deliteľ.**

```
int gcd(int a, int b)
{
    while (b != 0)
    {
        if (a > b)
            a = a - b;
        else
            b = b - a;
    }
    return a;
}
```



I Ciele programovania

- Zabezpečiť, že program správne vykonáva úlohy, na ktoré je určený.
- Zabezpečiť, že ľudia môžu jednoducho a príjemne program používať.
- Zabezpečiť, že program je zrozumiteľný, ľahko opraviťelný, a je možné ho zlepšiť bez veľkej námahy.
- Programy nám majú život robiť jednoduchším, nie zložitejším...

I Ako sa stať programátorom?

- Zvyčajná úloha programátora: **Vytvorit' riešenie problému využitím počítačového programu, ktorý môže by opakovane použitý pre rôzne výskyty toho istého problému.**
- Programátor musí:
 1. Porozumieť problému
 2. Vymyslieť a navrhnuť riešenie
 3. Vyjadriť riešenie v programovacom jazyku
- Najlepší programátori **PRAVIDELNE** robia chyby
- **Programujte po malých kúskoch, a každý kúsok si dôsledne overte, či je správny.**

2 Spôsob práce na tomto predmete

- **Prednášky (pondelok 9:00):**
zapájať sa, pýtať sa, počúvať, ...
písomný test (10%)
- **Cvičenia:**
konzultácie k úlohám
riešenie úloh, písanie programov (30%)
projekt (20%)
- Povinné minimum zo semestra 30 bodov
- **Skúška:**
písomný test (40%)

2 Vyučujúci

- Prednášky (pondelok 9:00):

Jozef Tvarožek

Pýtajte sa, chceme zodpovedať VŠETKY vaše otázky!

- Cvičenia:

Marián Potočný

Pýtajte sa, chceme vám pomôcť pochopiť a vyriešiť VŠETKY zadané úlohy!

- Emailová komunikácia – predmet správy:

[ZPrPr] Otazka ABC

Faculty of informatics and information technologies, Slovak University of Technology in Bratislava



Základy procedurálneho programovania 2016/2017

Mgr. Jozef Tvarožek, PhD.



Cieľom predmetu je získať základné zručnosti algoritmickej problematiky: použitie podmienok, cyklov, procedúr/funkcií, rekurzie a jednoduchých dátových štruktúr. Na praktickom riešení krátkych úloh získať skúsenosti s implementáciou algoritmickej problematiky v programovacom jazyku C. Prehľad základných konštrukcií programovacieho jazyka C: základné údajové typy, vstup, výstup, práca so súborami, dynamické pridelenie pamäte, ukazovatele.

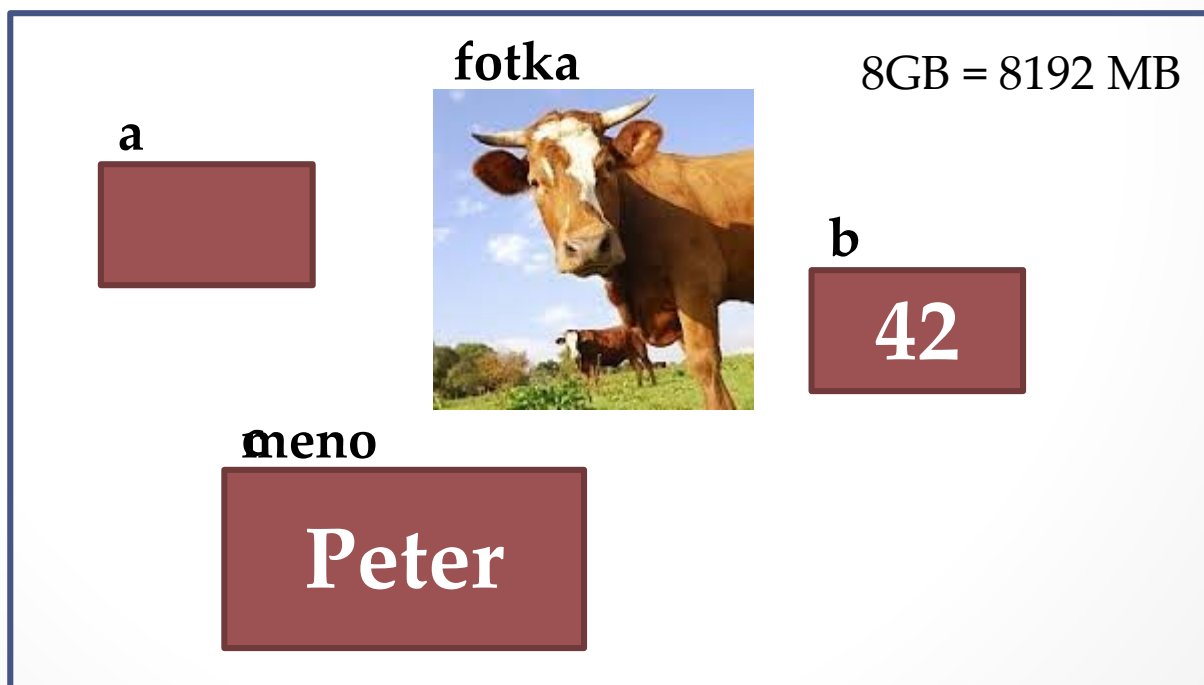
12 weeks · Recommended effort: 8-12 hours / week · 107 students

ENTER



3 Premenná – variable

- **Premenná** je pomenovaný priestor v pamäti pre uloženie dát



3 Premenná – variable

- Premenná môže obsahovať využiteľnú hodnotu (**inicializovaná**) alebo nemusí obsahovať využiteľnú hodnotu (**neinicializovaná**)
- Proces vyhradenia pamäti pre premennú a pomenovanie toho priestoru nazývame **vytvorenie** alebo **alokovanie** alebo **deklarovanie** premennej
- Proces naplnenia premennej hodnotou nazývame **priradenie** do premennej

3 Priradenie – assignment

- Premenná **vek**, naplnenie hodnotou 24

vek ← 24

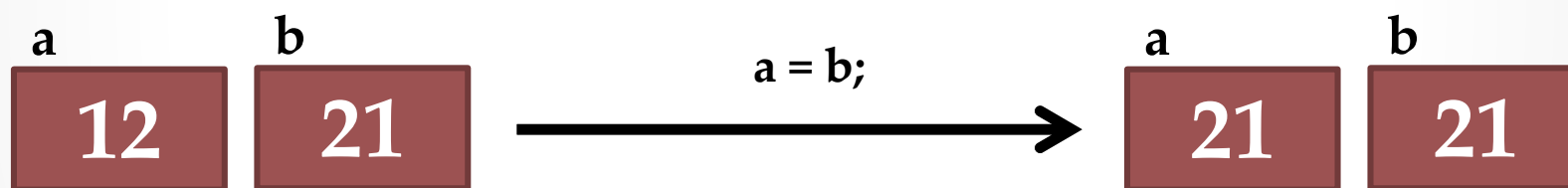
- **Inštrukcia** po vykonaní ktorej bude pamäť vyhradená pre premennú **vek** obsahovať hodnotu 24
- V rôznych programovacích jazykoch môže byť zapísaná rôzne:

Jazyk Pascal
vek := 24;

Jazyk C
vek = 24;

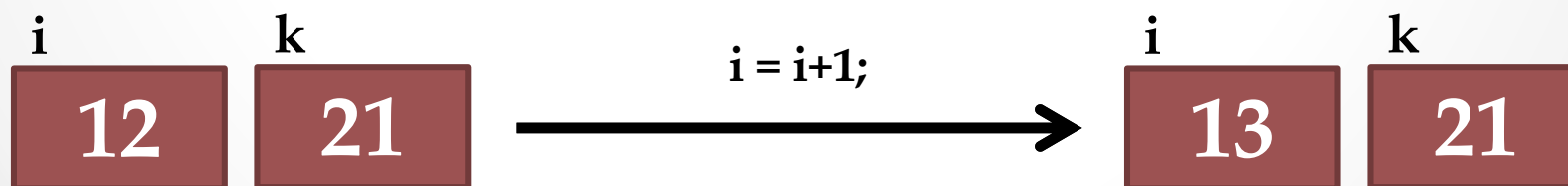
3 Priradenie – assignment

- $a \leftarrow b$ alebo (jazyk C) $a = b$;
inštrukcia po vykonaní ktorej bude pamäť vyhradená pre premennú **a** obsahovať hodnotu, ktorá bola v pamäti vyhradenej pre premennú **b**



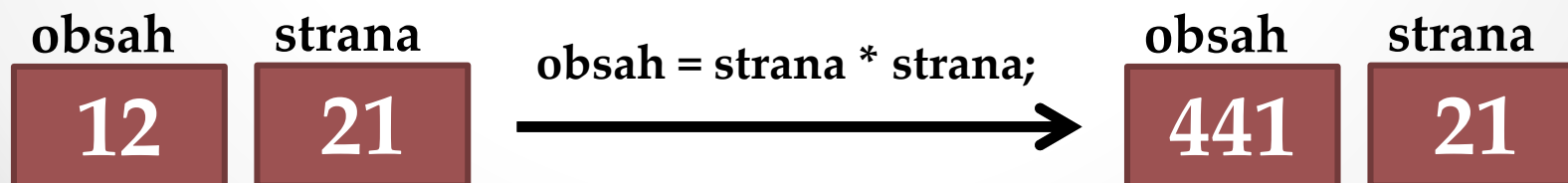
3 Priradenie – assignment

- $i \leftarrow i + 1$
- Rovnica (v jazyku C) $i = i + 1$;
sú **inštrukcie**, ktoré vykonajú:
 - 1) prečítajú hodnotu v pamäti vyhradenej pre premennú i
 - 2) procesor hodnotu zvýši o 1 a výsledok zapíše (uloží) do pamäte vyhradenej pre premennú i
- Skrátенý zápis: $i++$; je to isté ako $i = i + 1$;



3 Výraz (aritmetický) – Expression (arithmetic)

- **obsah** \leftarrow **strana** \times **strana**
- Rovnica v jazyku C **obsah = strana * strana;** sú **inštrukcie**, ktoré vykonajú:
 - 1) prečítajú hodnotu v pamäti vyhradenej pre premennú *strana*
 - 2) **vyhodnotia aritmetický výraz** $\text{strana} * \text{strana}$ procesor hodnoty vynásobí (operácia $*$)
 - 3) výslednú hodnotu súčinu zapíše (uloží) do pamäte vyhradenej pre premennú *obsah*



3 Výraz (aritmetický) – Expression (arithmetic)

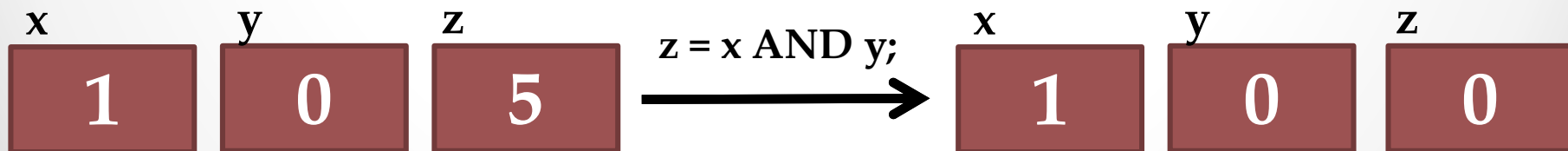
- Výpočet hodnoty výrazu, nazývame **vyhodnotenie výrazu**
- Vyhodnotenie aritmetického výrazu podlieha tradičných pravidlám aritmetiky
- Pravidlá poradia vyhodnocovania (precedencia operácií)
- zľava doprava, násobenie a delenie pred sčítaním a odčítaním
- Hodnota $x = a + b * c + d$ pre $a = 10$, $b = 20$, $c = 30$, $d = 40$?
- $10 + 20 * 30 + 40 = 10 + 600 + 40 = \mathbf{650}$

3 Operátor priradenia

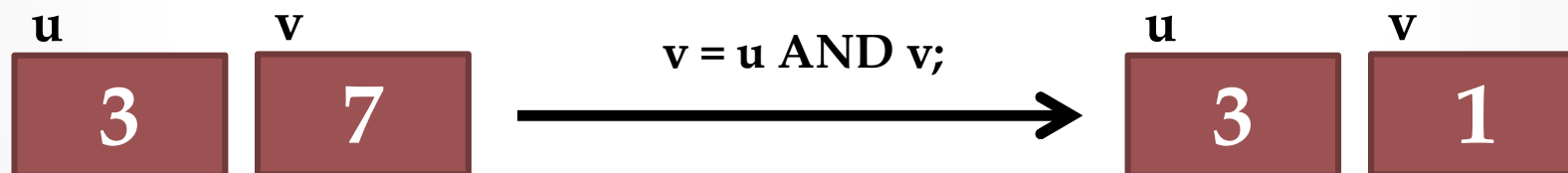
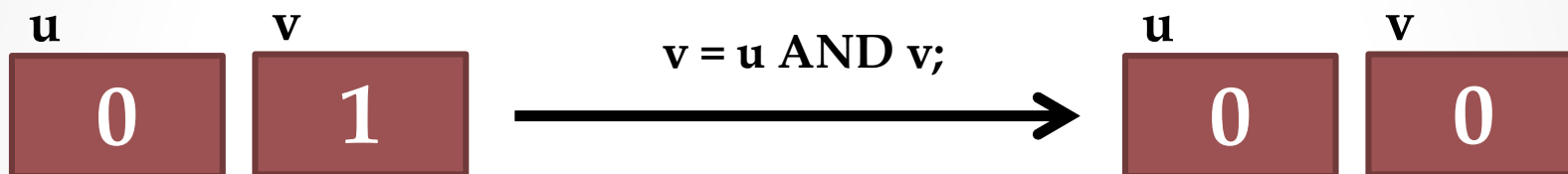
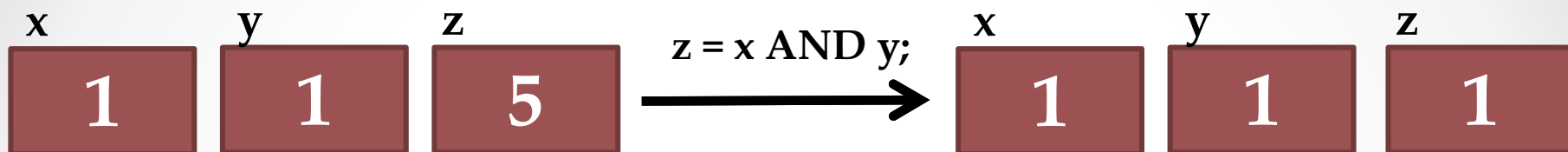
- Aj priradenie ($=$) je operácia, výsledok ktorej je priradená hodnota
- asociuje sa sprava
napr. priradenie $a = b = c$ je vlastne $a = (b = c)$
vykoná inštrukcie:
 - 1) priradí hodnotu premennej c do premennej b
 - 2) priradí výsledok operácie $b = c$ do premennej a
- Rôzne programovacie jazyky môžu mať rôzne precedencie operátorov a asociačné pravidlá

4 Výraz (logický) – Expression (logical)

- Výpočet binárnej (0/1) hodnoty
0=false (nepravda) inak je true (pravda) = 1
- Logické operátory AND, OR, NOT, EQUAL, >, <, ...
Hodnoty (premenných / medzivýsledkov) vo výraze nadobúdajú len hodnoty false (0) alebo true (1)
- $x \text{ AND } y$ znamená, že logický výraz je pravdivý (true), ak sú súčasne pravdivé výrazy x aj y .

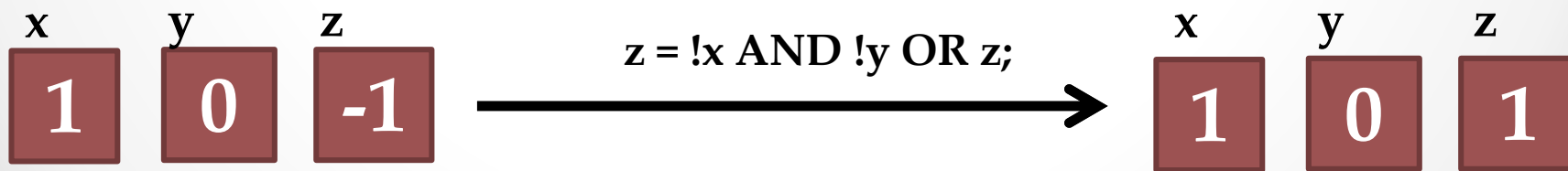


4 Výraz (logický) – Expression (logical)



4 Výraz (logický) – Expression (logical)

- Logický výraz **x OR y** je pravdivý (true), ak je pravdivý výraz x alebo je pravdivý y.
- Logický výraz **x EQUAL y** je pravdivý ak je hodnota x rovnaká ako hodnota y
- Logický výraz **NOT x** je pravdivý ak bola hodnota x nepravdivá, zapisuje sa aj ako ! (výkričník)
- Precedencia operátorov: NOT, <, >, EQUAL, AND, OR, ...



4 Výraz (logický) – Expression (logical)

- Ak je hodnota premennej nenulová, tak sa v logickom výraze vyhodnotí ako pravda (true):
0 je false, 1 je true, 47 je true, -3 je true, ...
- Programy sa väčšinou snažia vyhodnotiť logický výraz čo najefektívnejšie (vykonaním čo najmenej operácií), tzv. **skrátané vyhodnocovanie** (short-circuiting), používa sa v jazyku C

4 Výraz (logický) – Expression (logical)

- **Skrátené vyhodnocovanie** (short-circuiting)
AND je 0(false) ak aspoň jeden z výrazov je 0(false)
OR je 1(true) ak aspoň jeden z výrazov je 1(true)



- Pri **`x = x AND y AND z`** sa vykonajú inštrukcie:
1) načítanie hodnoty x, keďže x je nenulová,
pokračujeme 2) načítanie hodnoty y, keďže y je 0
vyhodnocovanie končí, lebo bez ohľadu na ďalšie
členy vo výraze je výsledok 0
- Hodnota **z** sa nenačíta a nebude sa vyhodnocovať
(napr. ak z je volanie funkcie, tak sa funkcia nezavolá!)

5 Podmienka (if / else) – Condition

- Program vykonáva príkazy postupne za sebou
- Niekedy chceme v špecifických prípadoch vykonať iné príkazy, napr.:

```
pocetDniRoku = 365;  
if (priestupnyRok)  
    pocetDniRoku++;
```

- Pri každom vykonaní programu:
 - 1) priradíme do premennej pocetDniRoku hodnotu 365
 - 2) ak hodnota premennej priestupnyRok je true (nenulová), tak hodnotu v premennej pocetDniRoku zvýšime o 1.
- Vykonanie programu môže vykonať rôzne inštrukcie podľa aktuálneho stavu premenných

5 Podmienka (if / else) – Condition

- **Príkaz vetvenia** obsahuje zvyčajne tri časti:
 - 1) kľúčové slovo **if**
 - 2) logický výraz (podmienku), ktorá sa vyhodnotí
 - 3) príkaz alebo skupinu príkazov, ktoré sa vykonajú **PRÁVE VTEDY** keď podmienka (2) je pravdivá.
- Rozšírenie o vetvu **else** – príkaz alebo skupina príkazov, ktoré sa vykonajú keď podmienka (2) je nepravdivá

```
if (priestupnyRok)
    pocetDniRoku=366;
if (!priestupnyRok)
    pocetDniRoku=365;
```

```
if (priestupnyRok)
    pocetDniRoku=366;
else
    pocetDniRoku=365;
```