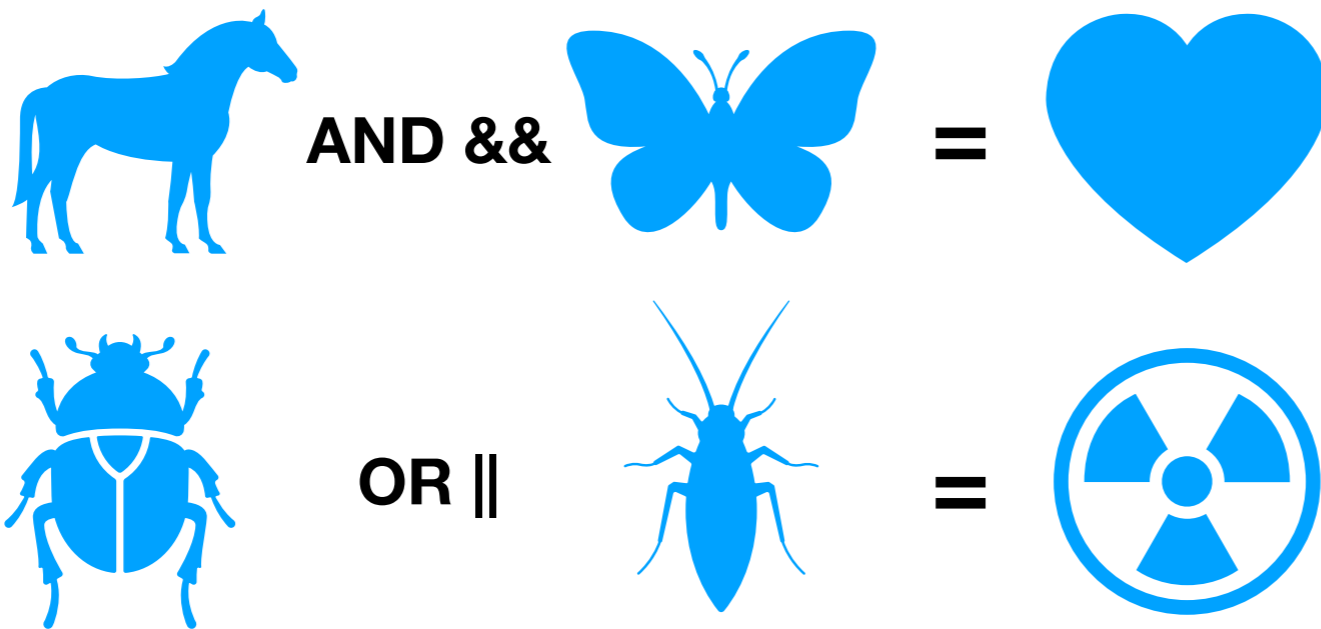


# ZAP-PM III.

Matej Gazda

# Motivácia

Využitie logických a aritmetických operátorov pri vetvení programu



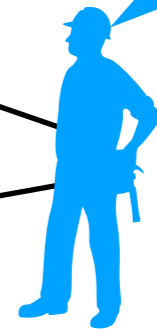
Použitie nových konštrukcií pri vetvení programu (okrem if-else aj ďalšie)

Veľa programovania

# For cyklus

```
for (int i=0; i<10; i++) {  
    printf ("%d\n", i);  
}
```

```
int i = 0;  
while (i < 10) {  
    printf ("%d\n", i);  
    // i++;  
    i = i+1;  
}
```



Rozdiel?

# Arithmetic operators

Operátor	Vysvetlenie	Príklad
+	Sčítanie	2+2
-	Odčítanie	2-2
*	Násobenie	2*2
/	Delenie	2/2
%	Získa zvyšok po delení	2%2
i++	Inkrementácia o 1 jedntoku, ekvivalent $i = i+1$	i++
i--	Dekrementácia o 1 jednotku, ekvivalent $i = i-1$	i--

# Relational operators

Operátor	Vysvetlenie	Príklad
<code>==</code>	Equal (rovnosť)	<code>5 == 5</code>
<code>!=</code>	Not equal (nerovné)	<code>5 != 4</code>
<code>&gt;</code>	Väčšie	<code>5 &gt; 4</code>
<code>&lt;</code>	Menšie	<code>5 &lt; 4</code>
<code>&gt;=</code>	Väčšie alebo rovné ako	<code>5 &gt;= 4</code>
<code>&lt;=</code>	Menšie alebo rovné ako	<code>5 &lt;= 4</code>

# Logical operators

Operator	Vysvetlenie	Príklad
&&	Musia platiť obe podmienky	A && B
	Musí platiť A alebo B	A    B
!	Negácia (napr. !False = True)	!(A && B)

# Ďalšie informácie

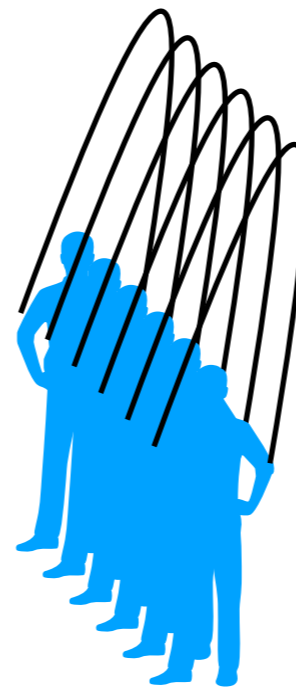
Funkcia	Vysvetlenie	Príklad
sizeof()	Navráti veľkosť premennej	sizeof(int)
?:	Podmienka ako if	

# Rekurzia

Rekurzia - proces opakovaného volania samého seba

Rekurzia v programovaní:

Funkcia zavolá samú seba

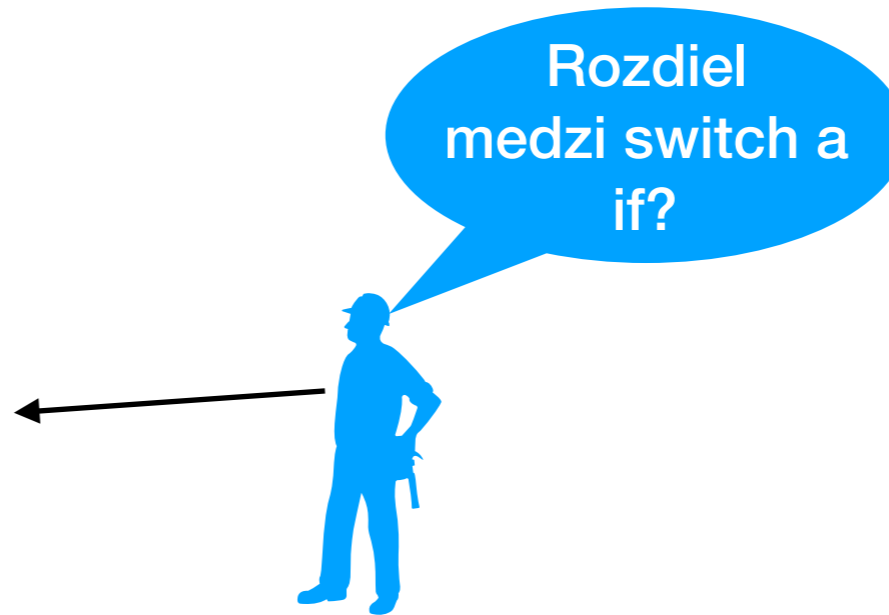


Úloha: Faktoriál pomocou rekurzie!



# Switch (vetvenie)

```
1. switch (expression)
2. {
3.     case constant1:
4.         // statements
5.         break;
6.
7.     case constant2:
8.         // statements
9.         break;
10.    .
11.    .
12.    .
13.    default:
14.        // default statements
15. }
```



Rozdiel  
medzi switch a  
if?

# Zadania



**Deadline, 11. 12. Týždeň**

**Zadanie musí mať aj istú náročnosť**



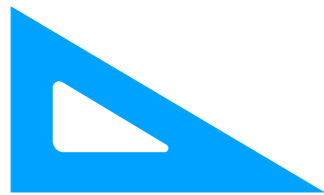
**Možnosť odkomunikovať so mnou úlohu zadania**

**Ak nemáte vlastný nápad, niečo Vám vymyslím**

# Úloha na prednášku

Prediskutujte a vytvorte funkcie, ktoré vypíšu nasledujúce postupnosti (do čísla N)

- 1, 2, 3, 4, 5, 6 .... N
- 1, 3, 5, 7, 9, 11 .... N
- 1, 5, 9, 13, 17, 21 ... N
- 1, 2, 4, 7, 11, 16 ... N
- 1, 2, 4, 8, 16, 32, ... N
- 1, 2, 6, 24, 120, ... N
- 1, 1, 2, 1, 2, 3, 1, 2, 3, 4, 1, 2, ... N
- 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59 ... N



# Úloha na prednášku

$3n+1$  problém

**Úloha:**

**Vezmi prirodzené číslo, pokiaľ je párne vydeľ ho 2, pokiaľ je nepárne vynásob ho tromi a pripočítaj jednotku. Postup opakuj, pokiaľ nedostaneš číslo 1.**

**Príklad:**

**7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1**

# Úloha na přednášce

Úloha:

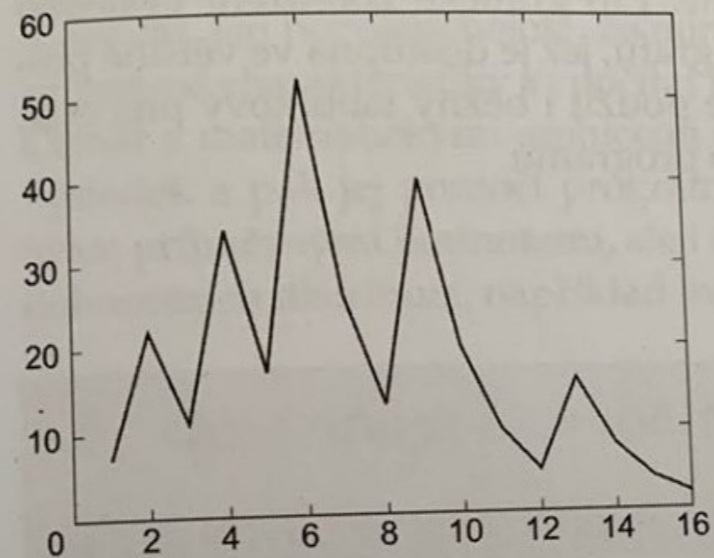
Vezmi prirodzené  
a prip

Príklad:

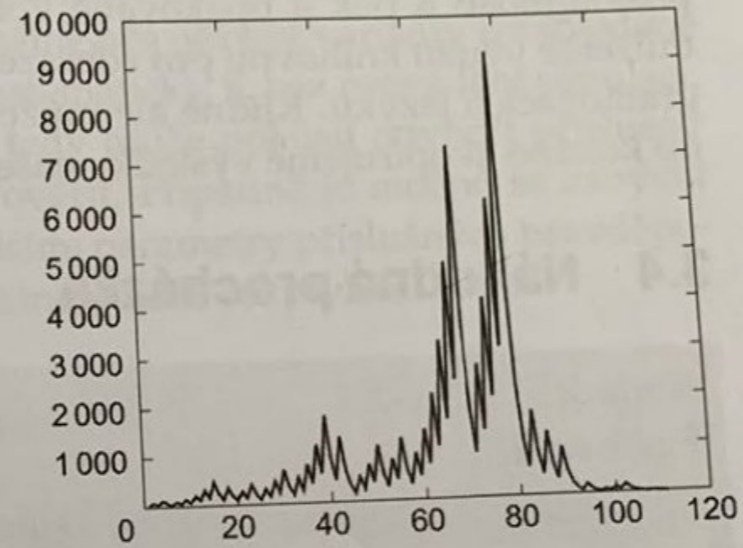
7, 22, 11, 34, 17, 5

A) konkrétní posloupnosti

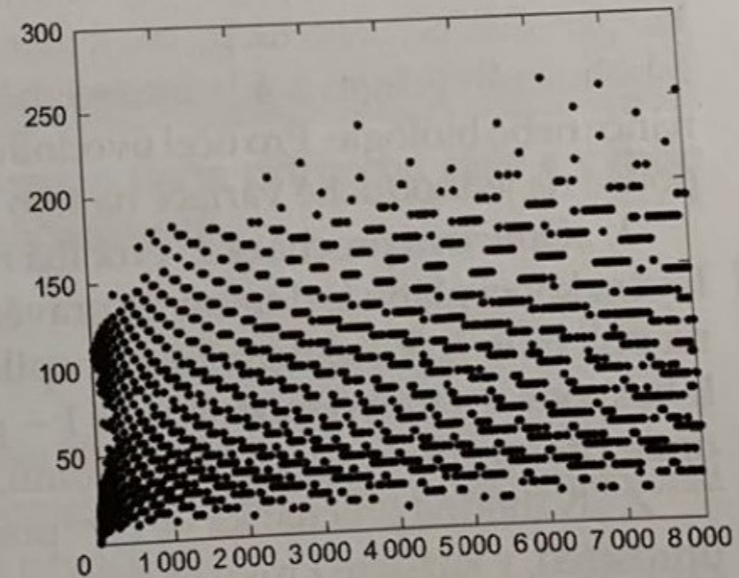
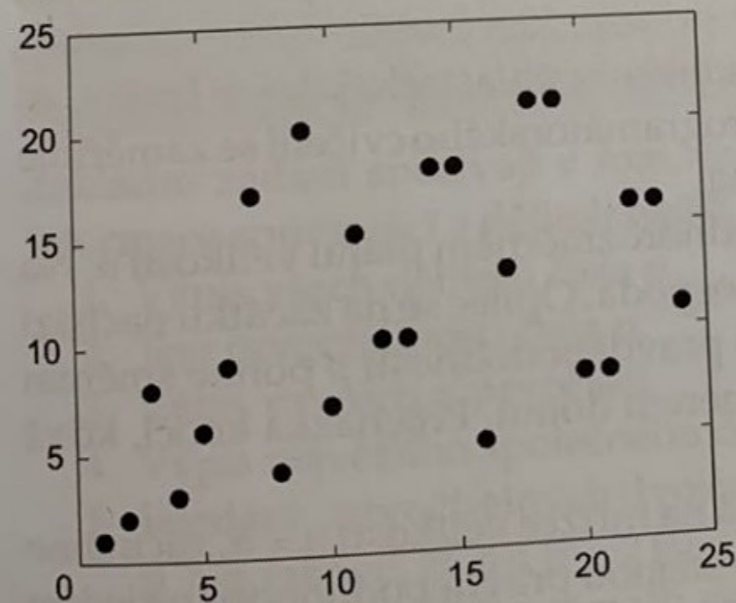
začínající číslem 7



začínající číslem 27



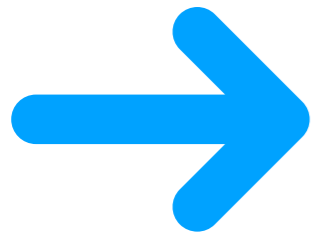
B) počet potřebných kroků



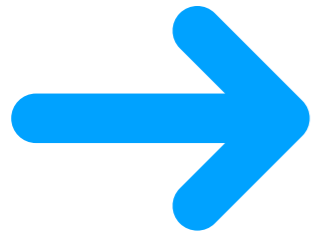
Obrázek 3.1: Collatzův problém – grafy

ynásob ho troma  
slo 1.

# Úloha na prednášku



**Napíšte program, ktorý načíta zo vstupu znak a rozhodne, či sa jedná o písmenko alebo číslo. Rozhodnite, či je písmenko veľké alebo malé (A vs a) a či je číslo párne alebo nepárne**



**Napíšte program, ktorý bude načítavať znaky z klávesnice až po '\n' (Enter) a následne vypíše počet zadaných malých a veľkých písmen a počet číslic. Ostatné znaky ignorujte.**

# Úlohy na prednášku a cvičenie

## Úloha futbalová miniliga 2 proti 2:



- Vygenerovať dve náhodné čísla pre každého hráča (2 a 2)
- Opakovať tento postup 90x ( 90 minút vo futbale)
- V prípade, že:



1. Oba tímy budú mať súčet náhodných čísel svojich hráčov  $\geq 100$  pridať bod obom tímom
2. Jeden tím bude mať súčet väčší ako 100 a druhý menší, pridať bod iba tomu tímu
3. Ak budú mať obe tímy menší súčet čísel ako 100, neurobiť nič
4. Výsledok zapísať v tvare: Tím <názov tímu> zvíťazil nad tímom <názov tímu> x:x. (V prípade, že nastane remíza treba tiež niečo vypísať)

