

# Programujeme lépe a radostněji

© Leonard Wallezský 2003

# Uživatelský vstup

- Funkce *input*
- Použití:
  - $\langle \text{proměnná} \rangle = \text{input}(\langle \text{text} \rangle)$
  - Matlab vypíše text na obrazovku, čeká na uživatelský vstup a pak jej uloží do proměnné
- Např:  
 $a = \text{input}(\text{'Zadej číslo:'})$

# „Kontrolní výstup“

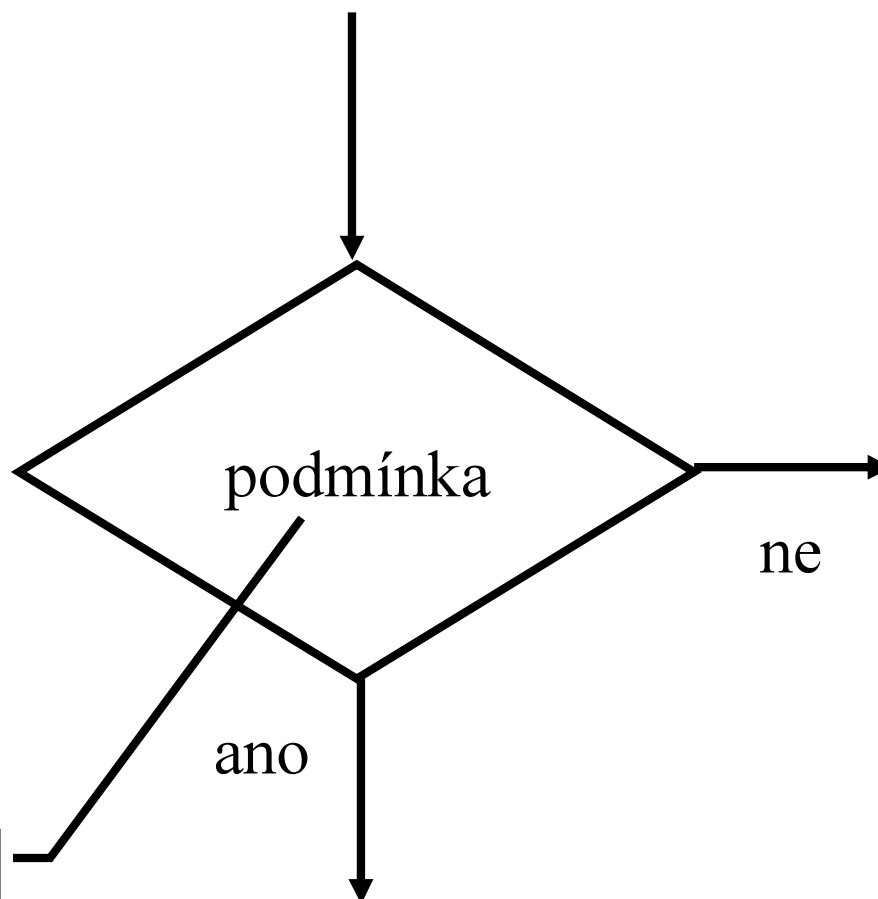
- Vždy, když modifikuji proměnnou, MATLAB změnu vypíše jako potvrzení
- Zabráním tomu, když na konec řádku umístím středník (bodkočiarku)
- Srovnej:
  - $x=2*56$
  - $x=2*56;$

# Modifikace příkladu s grafy

- Upravte program **grafy** tak aby:
  - neprováděl kontrolní výstupy jednotlivých výpočtů
  - načetl krok pro generování řady pro vektor  $x$ 
    - $k = \text{input}(\text{'Zadej krok:'});$
    - $x = 0:k:10;$
  - načetl i horní a dolní hranici intervalu
    - použijte proměnné  $h$  a  $d$

# Větvení algoritmu

rozhodování

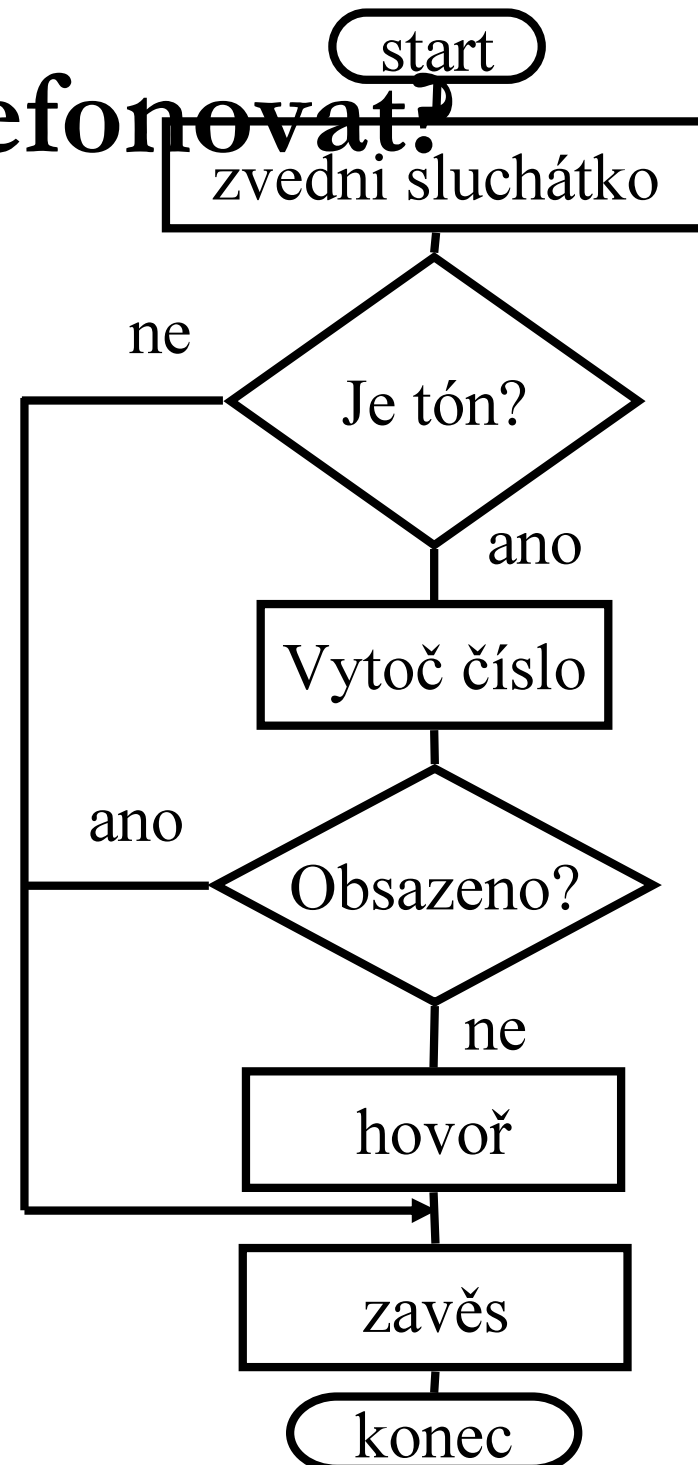


při splnění podmínky se pokračuje větví **ano**, při nesplnění větví **ne**

# Jak telefonovat?

Úkol:

Napsat algoritmus, podle kterého bude moci telefonovat i jedinec, který telefon nikdy neviděl.



# Větvení programu

- Klíčové slovo *if*
  - Znamená podmíněný příkaz
  - Syntaxe:  
*if* <podmínka>  
    <blok příkazů 1>

Jestliže je podmínka splněná, provede se blok příkazů 1, pokud ne, provede se blok příkazů 2

*end*

Část začínající *else* je nepovinná a nemusí se vždy uvádět (neúplný podmíněný příkaz)

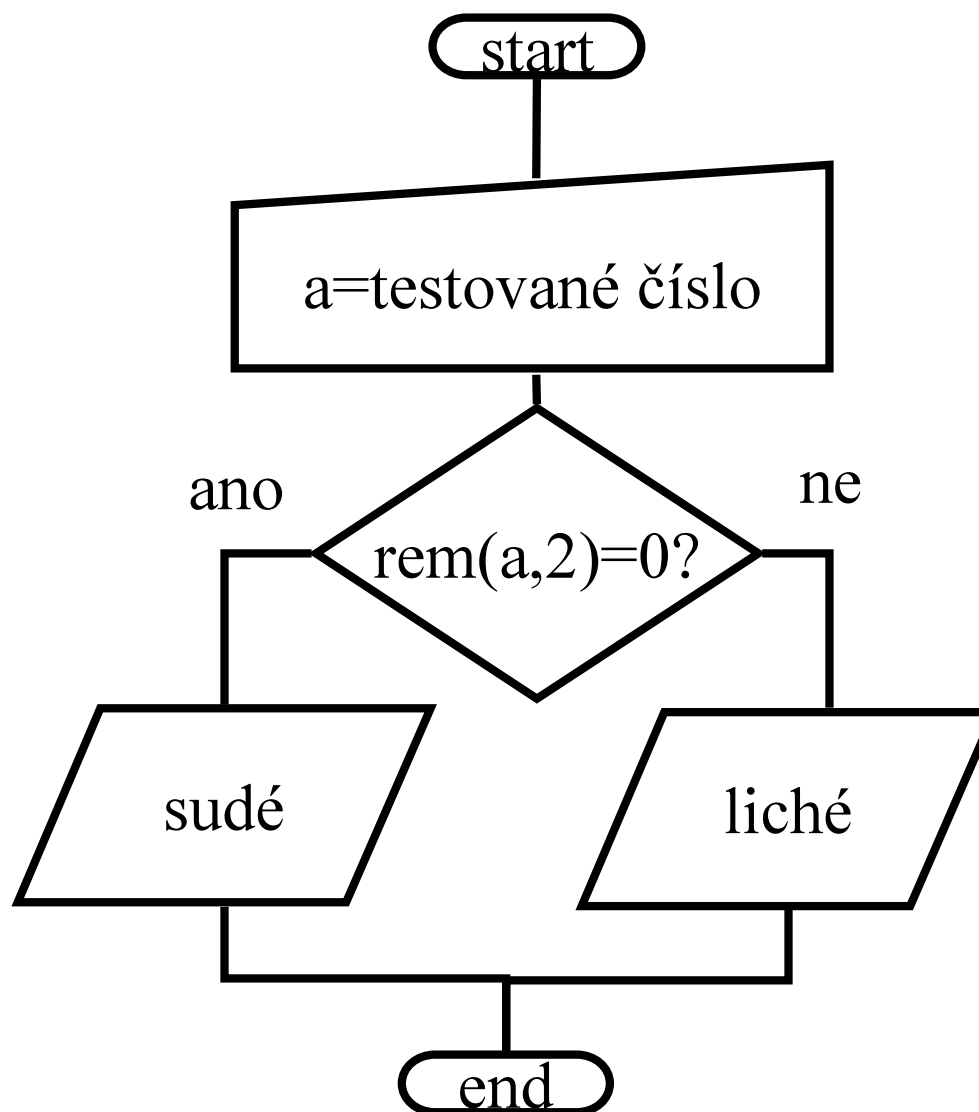
# Podmínky

- Symboly pro porovnávání
  - $=$  - ekvivalence (je rovno) – např.  $a = 3$
  - $\neq$  - neekvivalence (není rovno) např.  $b \neq 0$
  - $>$  - je větší (např.  $a > b$ )
  - $<$  - je menší (např.  $b < c$ )
  - $\geq$  - větší nebo rovno (např.  $c \geq 100$ )
  - $\leq$  - menší nebo rovno (např.  $d \leq 0$ )
- Je-li podmínek více, lze je spojovat logickými spojkami
  - $\&$  - logický součin, podmínky musí platit současně  
např.  $(a > 0) \& (a < 10)$
  - $|$  - logický součet, musí platit alespoň jedna podmínka  
např.  $(a > 2) | (a < -2)$
  - $\sim$  - negace, obrací pravdivostní hodnotu  
např.  $\sim (a = 0)$

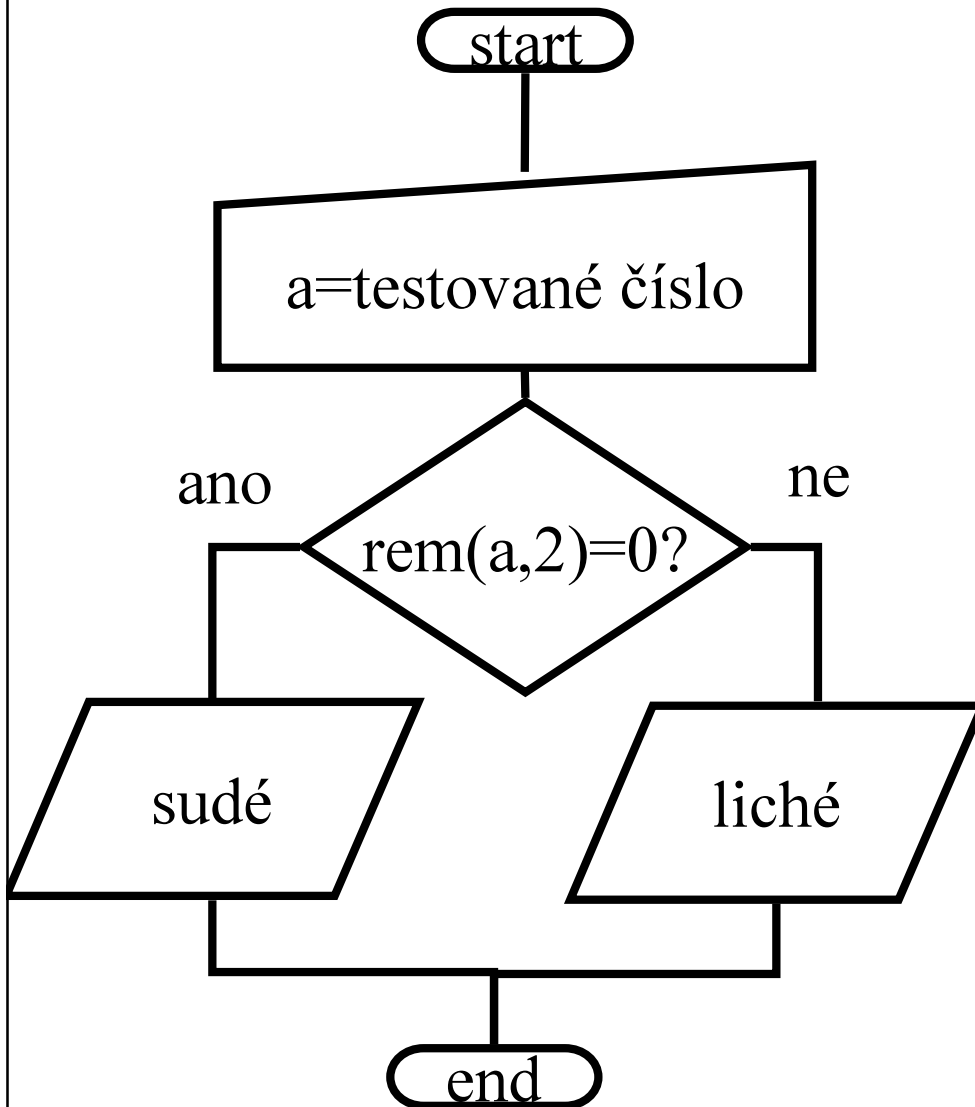


# Příklad na větvení programu

- Úkol: Napsat program, který rozhodne, jestli je zadané číslo sudé, nebo liché
- Řešení:
  - Využijeme funkce `rem`
  - Vývojový diagram:



# Přepis do MATLABu



```
a=input('Zadej cislo:');  
if rem(a,2)==0  
    disp('sude')  
else  
    disp('liche')  
end
```

- založíme nový m-file
- přepíšeme program
- uložíme pod názvem **porovnej.m**
- spustíme z MATLABu

# Další úkol

- Napište program, kterému zadáme dvě čísla – dělence a dělitele. Program rozhodne, zda je dělenec beze zbytku dělitelný dělitelem.
- Napište program, který seřadí dvě čísla, zadaná na vstupu podle velikosti.
- Upravte program tak, aby správně reagoval při rovnosti čísel na vstupu.