

# PROGRAMY PRO GIS

Formovat/formulovat problém pro aplikaci v počítači

Fungování GIS programů na základní úrovni - "uvažovat" jako počítač

## Jak počítače řeší problémy

procesor central processing unit - CPU

sekvence jednodušších kroků/operací = **algoritmus**

zvláštnost pro GIS: práce s prostorovými daty

## **Jak počítače vyjadřují okolní svět - datová modelace**

vybrat informaci, která je důležitá pro řešení problému

rozhodnout jak ji uložit v počítači

proces vytváření počítačové reprezentace reálného jevu/objektu = datová modelace

### Obecné kroky procesu datové modelace:

1. **model aplikačního prostředí** – není počítačový model

pro GIS: výchozí model aplikačního prostředí byl statický 2D = mapa

dva hlavní typy: topografická mapa, tématická mapa

informace dvojího druhu: kde, co → počítač musí pracovat s oběma druhy informace  
současný vývoj v GIS: model prostředí pro to, co mapa neumožňuje - časová změna, výškové variace, nemapové pohledy (např. uliční).

Mapový způsob je stále hojně využíván pro aplikace prostorových dat

2. **konceptní počítačový model** – nezávislý na druhu HW a SW

pro GIS: dva vhodné konceptní počítačové modely (datové modely) – vektor and raster  
tj. obecné způsoby uložení mapy do počítače

3. **logický počítačový model** – nezávislý na druhu HW, pro určitý SW

pro GIS: tvorba vhodného způsobu (algoritmus) přenesení prostorových dat z mapy do počítače

reprezentace datového modelu v konkrétním programovacím jazyku (např. C++)

4. **fyzický počítačový model** – závislý na konkrétním HW i SW

konkrétní programové soubory uložené po kompilaci v určitém typu počítače (např. vektorová data jako soubor v programu ArcGIS)

# ALGORITMY OPERACÍ S VEKTORY

Operace hledají odpovědi na otázku danou uživatelem při analýze.

Způsob nalezení odpovědi na konkrétní dotaz nemá obecnou použitelnost, ale mění se v závislosti na konkrétním případě (např, zda jeden objekt je vnořen do druhého nebo jaká je vzdálenost dvou objektů).

## ALGORITMUS PRO ZJIŠTĚNÍ PRŮSEČÍKU DVOU ČAR

jak zjistit zda se dvě čáry protínají - netriviální otázka pro počítač

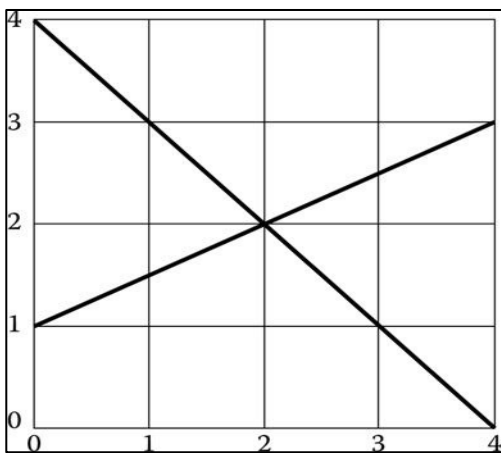
čára je v GIS reprezentována řadou bodů – sestavou navazujících úseček

### Zjištění průsečíku dvou úseček

Příklad:

grafická prezentace pro člověka:

prezentace pro počítač: osm čísel



		X	Y
Úsečka 1	začátek	0	1
	konec	4	3
Úsečka 2	začátek	0	4
	konec	4	0

Rovnice přímky

$$y = a + bx$$

směrnice

$$b = \frac{y_{\max} - y_{\min}}{x_{\max} - x_{\min}}$$

$$a = y \text{ pro } x = 0$$

$$y_1 = 1 + 0,5x_1$$

$$y_2 = 4 - x_2$$

průsečík existuje:  $x_p = 2 \quad y_p = 2$

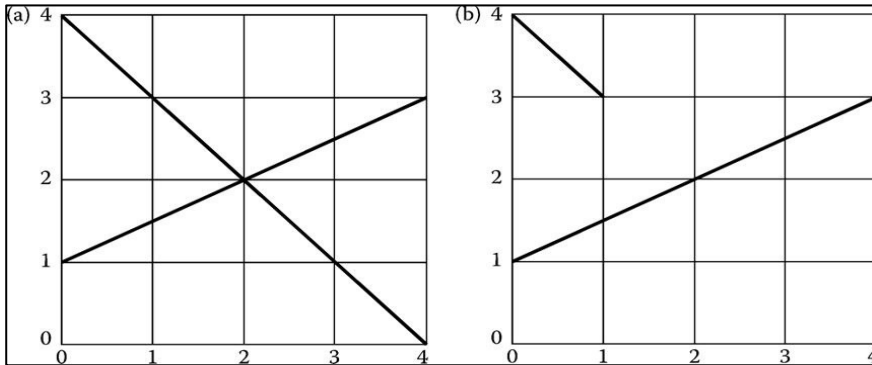
odpověď na otázku získaná tímto algoritmem: dané úsečky se protínají

Algoritmus musí být **obecný**

určení souřadnic průsečíku pro obecný tvar přímek:

$$y_p = a_1 + b_1 x_p = a_2 + b_2 x_p$$

Ale neřeší tuto situaci:



Úsečka 2 má definiční obor od  $x = 0$  do  $x = 1$ , průsečík má  $x_p = 2$ , leží mimo úsečku a proto se úsečky neprotínají  
Kriterium zda bod  $x_p$  leží uvnitř definičního oboru  $\langle x_1, x_2 \rangle$

$$(x_1 - x_p) \cdot (x_p - x_2) \geq 0$$

Ještě dva zvláštní případy:

a) úsečky jsou rovnoběžné

$b_1 - b_2 = 0 \rightarrow$  dělení nulou  $\rightarrow$  na začátku ověřit rovnost směrnic

b) jedna úsečka je vertikální

$x_{\max} - x_{\min} = 0 \rightarrow$  dělení nulou při výpočtu směrnice

$x_p = x_{\min} = x_{\max} \rightarrow y_p$  získáme přímo z rovnice přímky:  $y_p = a + b x_p$

Zbývají rozřešit dva problémy:

1) pokud jsou úsečky shodné po celé délce nebo po části jedné je třeba definovat zda odpověď bude: neprotínají se nebo: protínají se ve více bodech

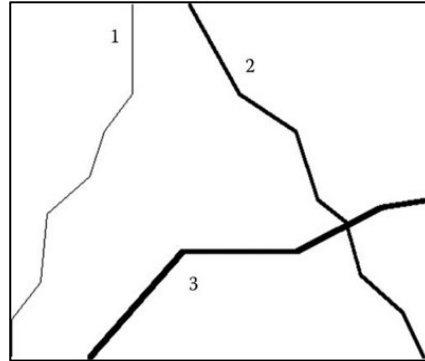
2) algoritmus může dát nesprávnou odpověď hlavně, když souřadnice bodů budou vyjádřeny hodně velkými nebo hodně malými čísly

## ČÁRY Z VÍCE ÚSEKŮ

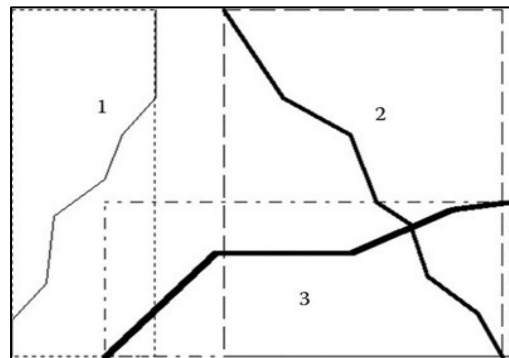
Možnost: prověřovat navzájem všechny úsečkové segment obou čar dokud se nenajde jeden pár, který se protíná

Nevýhoda: dlouhá doba, mnoho zbytečných testů

Nejdříve: zjistit zda se čáry vůbec mohou protínat



Metoda nejmenšího opsaného obdélníku



NOO je definován čtyřmi čísly – nejmenší a největší hodnoty souřadnic x a y ze všech bodů čáry

program na stanovení těchto 4 hodnot pro každou čáru

podmínka vyločení protnutí dvou čar:  $(Xmin)_1 > (Xmax)_2$

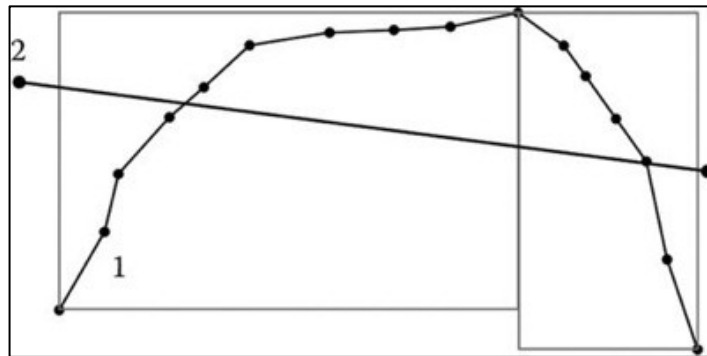
algoritmus:

```
if  $Xmin_1 > Xmax_2$  OR  $xmax_1 < xmin_2$  OR  $ymin_1 > ymax_2$  OR  $ymax_1 < ymin_2$ 
then
  NOO se neprotínají
else
  NOO se protínají
```

protínání NOO ještě neznamená, že se čáry protínají, ale test sníží počet možných případů

Další testovací postup:

- a) konstrukce NOO pro každý úsečkový segment čáry
- b) rozdělit čáru na monotonní úseky – trvalů růst nebo pokles hodnot X a Y, v každém takovém úseku se dvě čáry mohou protnout nejvýše jednou



#### OBECNĚ O ALGORITMU:

- 1) musí řešit všechny možné situace včetně speciálních případů
- 2) měl by být efektivní – s co nejmenším počtem výpočtů a výpočty provádět rychle. To může obsahovat i procesy zdánlivě navíc, ale šetří celkový čas (např. stanovení a uložení NOO pro každou čáru do databáze pro dělání detailních výřezů z dat)