



# Kartografické modelování

## II – Mapová algebra – obecné základy a lokální funkce

jaro 2014

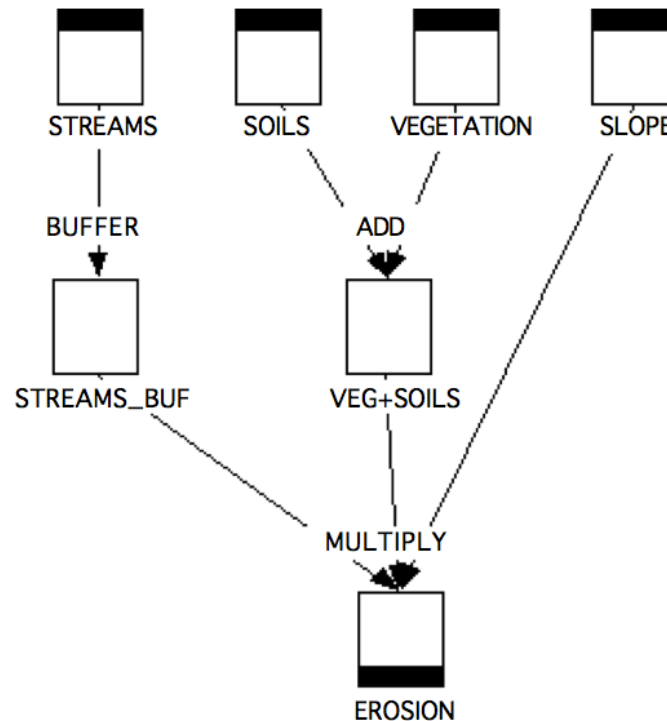
**Petr Kubíček**

**kubicek@geogr.muni.cz**

**Laboratory on Geoinformatics and Cartography (LGC)  
Institute of Geography  
Masaryk University  
Czech Republic**

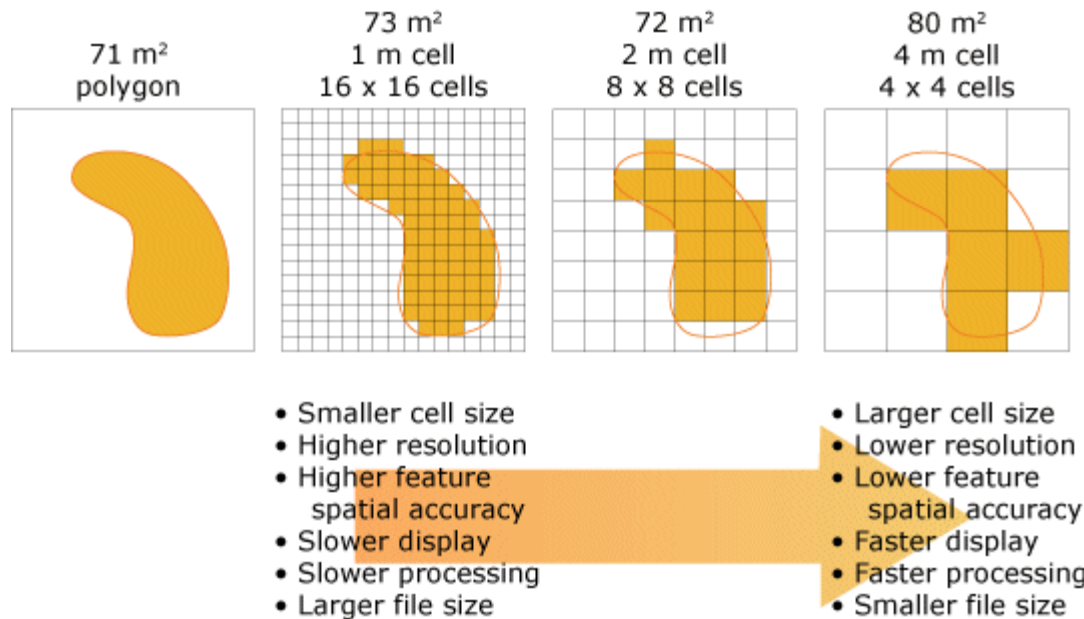
# Historie – mapová algebra

- Překryt, pravidelné dělení území
- Dana Tomlin, Berry
- Vrstva, výraz, funkce
- Flowchart diagram



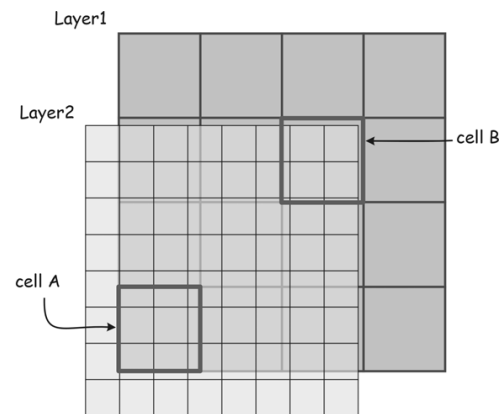
# Předpoklady – podmínky užití

- Pravoúhlá soustava čtvercových buněk
- Kategorie, bool, celá čísla, reálná, vektory
- No data



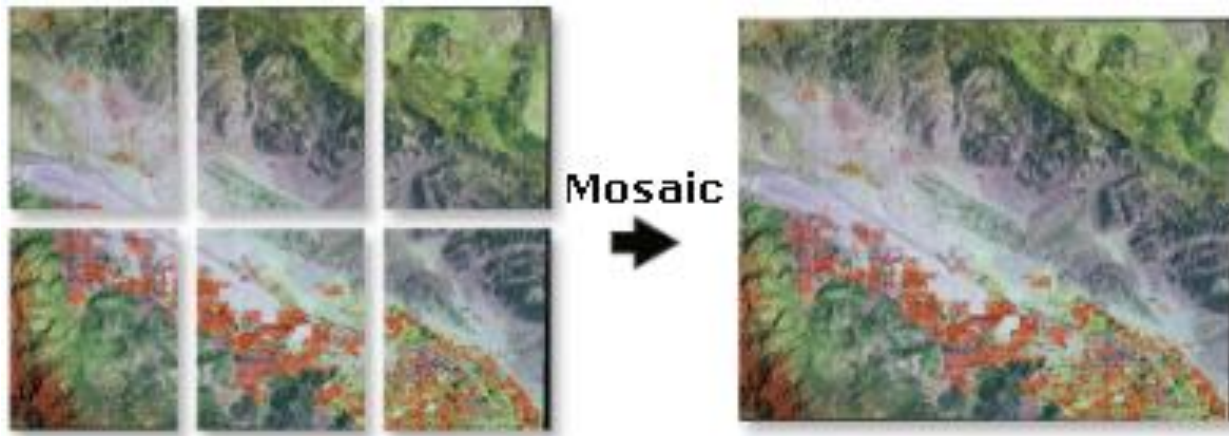
# Změna velikosti buňky - resampling

- **Resamplování – harmonizace buněk (velikost, poloha)**
  - Nejbližší soused (nearest 1) – ztráta, diskrétní
  - Převládající (majority 1) – zhlazení, diskrétní
  - Bilineární (4) – zhlazení, souvislá
  - Kubická konvoluce (16) – zhlazení, souvislá.



# Spojení gridů

- **Spojení gridů - merge, mosaic**



# ASCII to Grid

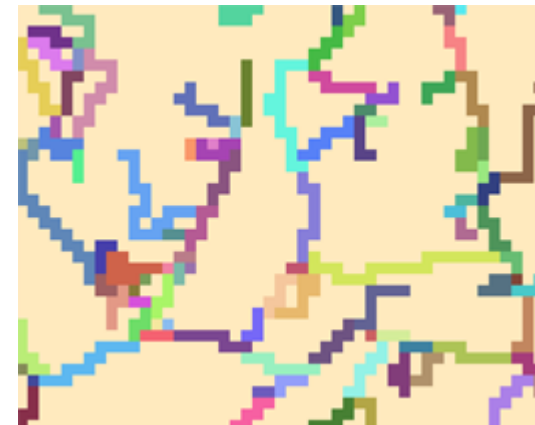
- NCOLS xxx
- NROWS xxx
- XLLCORNER xxx
- YLLCORNER xxx
- CELLSIZE xxx
- NODATA\_VALUE xxx
- row 1
- row 2
- .
- .
- row n

# Konverze mezi vektorem a rastrem

**Vstup (vektor)**



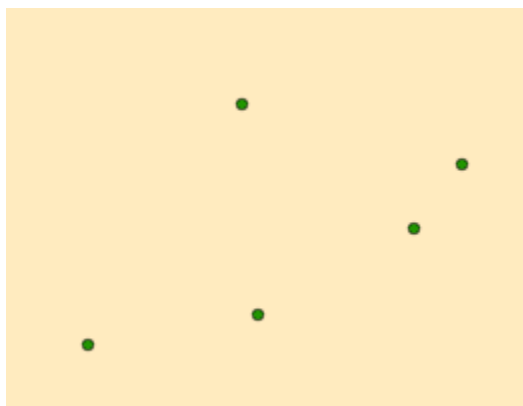
**výstup (rastr)**



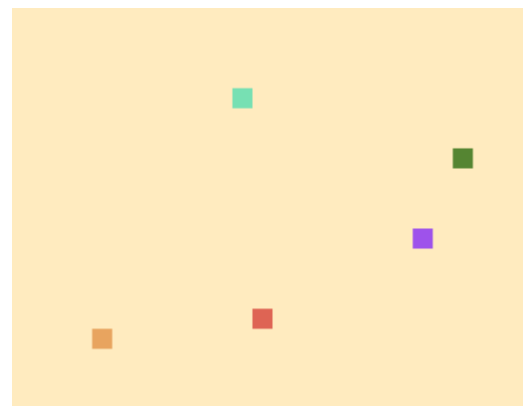
**Kartografické modelování**



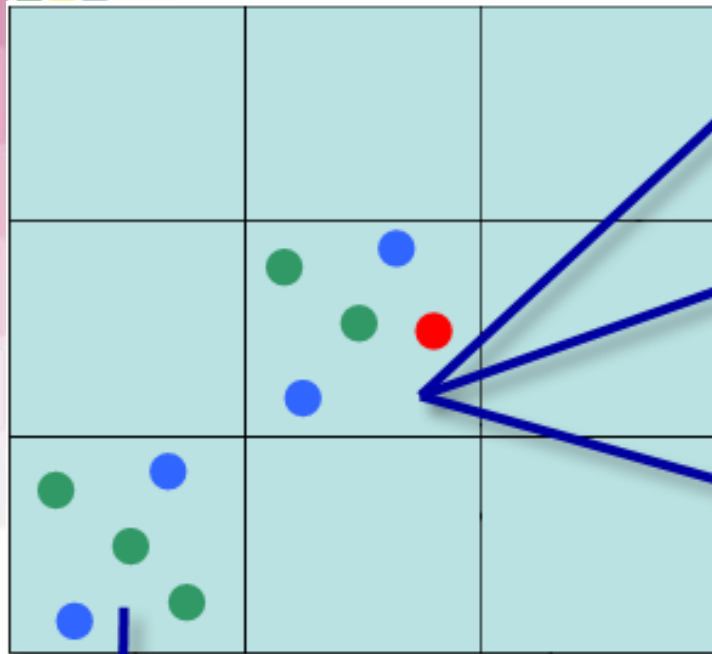
**Vstup (vektor)**



**výstup (rastr)**

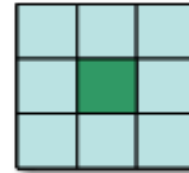






**FID | Attribute**

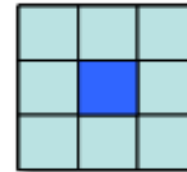
1	Green
2	Red
3	Blue
4	Blue
5	Green



Field = Attribute  
 Method = MOST\_FREQUENT  
 Priority = NONE  
 Outcome = Green  
 Reason = Lowest FID

**FID | Attribute | PriorityFID**

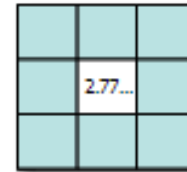
1	Green	1
2	Red	1
3	Blue	1
4	Blue	3
5	Green	2



Field = Attribute  
 Method = MOST\_FREQUENT  
 Priority = PriorityFID  
 Outcome = Blue  
 Reason = Highest priority

**FID | ValueFID**

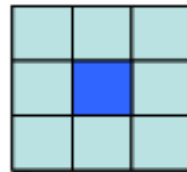
1	1
2	8
3	5
4	3
5	2



Field = ValueFID  
 Method = STANDARD\_DEVIATION  
 Priority = Ignored  
 Outcome = 2.774887323379517  
 Reason = Priority field is only used with MOST\_FREQUENT

**FID | Attribute | PriorityFID**

1	Green	1
2	Blue	2
3	Blue	2
4	Green	1
5	Green	2

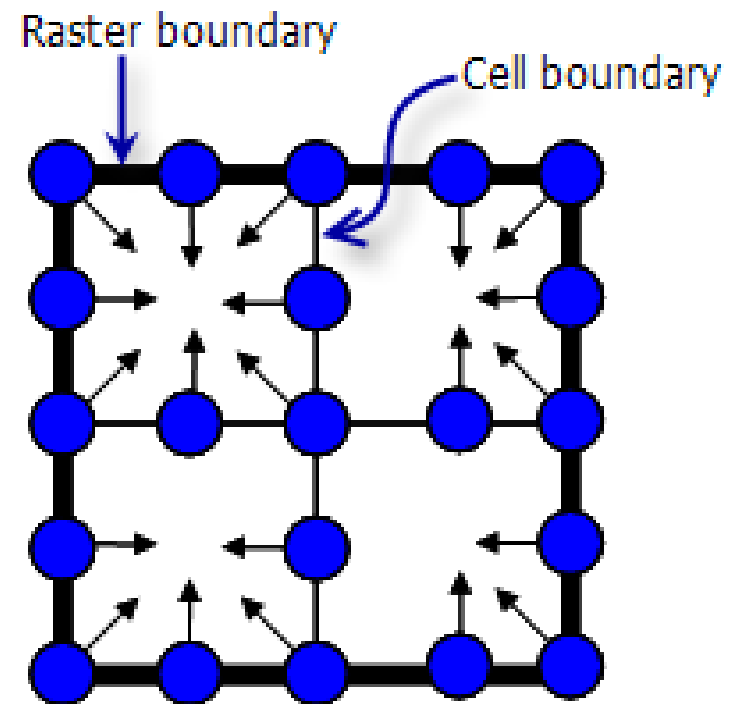


Field = Attribute  
 Method = MOST\_FREQUENT  
 Priority = PriorityFID  
 Outcome = Blue  
 Reason = Highest priority

## Kartografické modelování

# Bod na hranici rastru či buňky

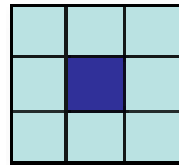
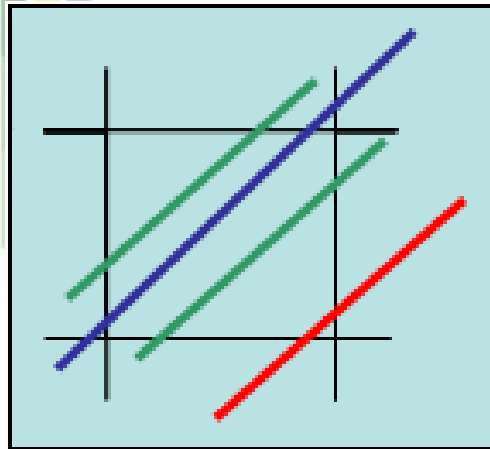
- Umístění – viz grafika
- Pro prvek na hranici užijte buňku nad prvkem, pokud tam je.
- Pro prvek na hranici užijte buňku nalevo, pokud tam je.



## Nástroj Polyline to Rastr

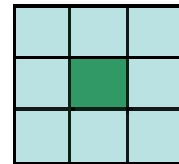
### Metody:

- MAXIMUM\_LENGTH
- MAXIMUM\_COMBINED\_LENGTH
- **Priority**



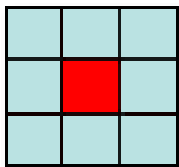
<u>FID</u>	<u>Attribute</u>
1	Green
2	Blue
3	Green
4	Red

Field = Value  
 Method = MAXIMUM\_LENGTH  
 Priority = NONE  
 Outcome = Blue  
 Reason = Longest length



<u>FID</u>	<u>Attribute</u>
1	Green
2	Blue
3	Green
4	Red

Field = Value  
 Method =  
 MAXIMUM\_COMBINED\_LENGTH  
 Priority = NONE  
 Outcome = Green  
 Reason = Length of two green



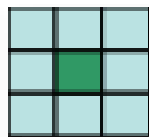
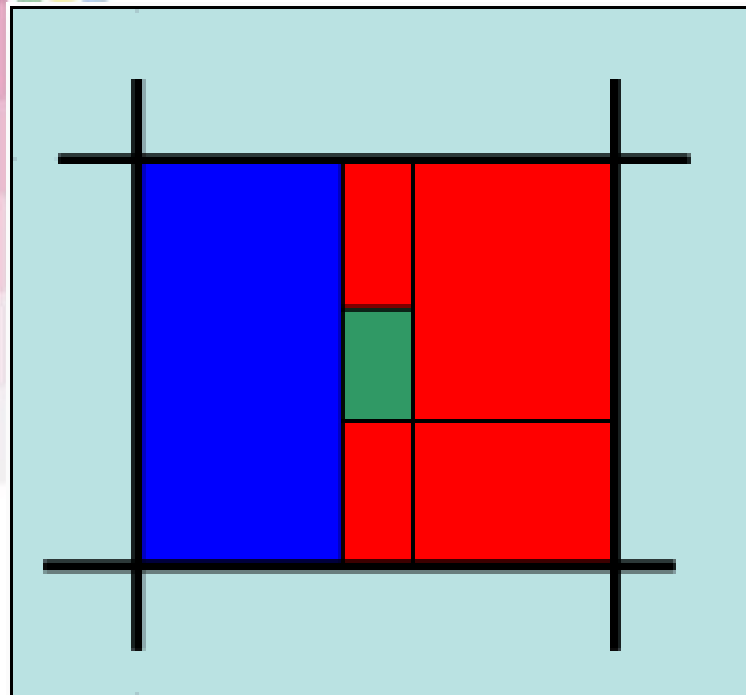
<u>FID</u>	<u>Attribute</u>	<u>PriorityFID</u>
1	Green	1
2	Blue	1
3	Green	2
4	Red	3

Field = Value  
 Method = MAXIMUM\_LENGTH  
 Priority = PriorityFID  
 Outcome = Red

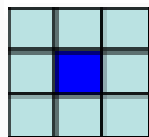
## Kartografické modelování

## Tři základní metody:

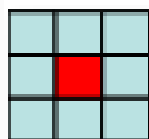
- **CELL\_CENTER** - hodnota nacházející se ve středu buňky.
- **MAXIMUM\_AREA** - největší souvislá plocha v buňce.
- **MAXIMUM\_COMBINED\_AREA** - největší sečtená plocha v buňce (i z více nesouvisejících ploch).
- Respektovány pravidla hranice
- Priorita podle FID



CELL\_CENTER



MAXIMUM\_AREA



MAXIMUM\_COMBINED\_AREA

<u>FID</u>	<u>Attribute</u>
1	Blue
2	Red
3	Green
4	Red
5	Red
6	Red

# Jazyk mapové algebry

- Nástrojů mapové algebry je možné využívat pomocí speciálního jazyka (jazyka mapové algebry).
- Jedná se o jednoduchý **programovací jazyk**
- navržený speciálně **pro popis analýz prostorového modelování** nad rastrovou reprezentací.
- Jeho syntaxe se produkt od produktu liší, ale princip zůstává stejný.



# Struktura jazyka MA

Mapová algebra používá **objekty**, **činnosti** a **kvalifikátory činnosti**. Ty mají obdobné funkce jako **podstatná jména**, **slovesa** a **příslovce**.

- **Objekty** slouží k uložení informací, nebo jsou to vstupní hodnoty. Jako objekty se používají rastry, tabulky, konstanty, ...
- **Činnosti** jsou příkazy jazyka (**operátory a funkce**) - vykonávají operace na objektech:
  - **Operátory** jsou obvyklé matematické, statistické, relační a logické operátory (+, -, \*, /, >, <, >=, <=, <>, mod, div, and, or, not, ...).
  - **Funkce** mapové algebry se dělí na lokální, fokální, zonální a globální.





# Operátory – základní pravidla použití

- Pokud v příkazu používáme **operátory**, musí být od objektů **odděleny mezerou z obou stran**.
- **Pořadí** provádění **operací** definovanými operátory je určeno pomocí tzv. **hodnoty nadřazenosti** (precedence value), která je přiřazena každému operátoru. Čím vyšší hodnota nadřazenosti, tím má operátor vyšší prioritu.
- Pokud operátory použité v příkazu mají stejnou hodnotu nadřazenosti, pak se operace vykonává „**zleva doprava**“. Nejdřív se vykoná operace daná operátorem, který se nachází víc nalevo od operátoru se stejnou hodnotou nadřazenosti.
- Pokud si chceme stanovit vlastní pořadí provádění operací, jednoduše **použijeme kulaté závorky** (tak jak se tomu běžně děje v klasické algebře).
- Pracujeme-li s **funkcemi**, všechny mají **stejnou hodnotu nadřazenosti**.
- Není-li pořadí určeno závorkami, vykonávají se operace postupně zleva doprava.

# Operace na jedné a více vrstvách

- **Z hlediska počtu zpracovávaných vrstev lze operace mapové algebry dělit na operace s jednou nebo více vrstvami.**
  - **Na jedné vrstvě** jsou to nejčastěji skalární operace jako je připočítávání konstanty, násobení, atp. Jako příklad může posloužit tvorba 2x převýšeného DMR pro vizualizaci ve 3D.
  - **Na více vrstvách jsou** to operace jako sčítání vrstev, které se vykonávají s prostorově odpovídajícími si buňkami.

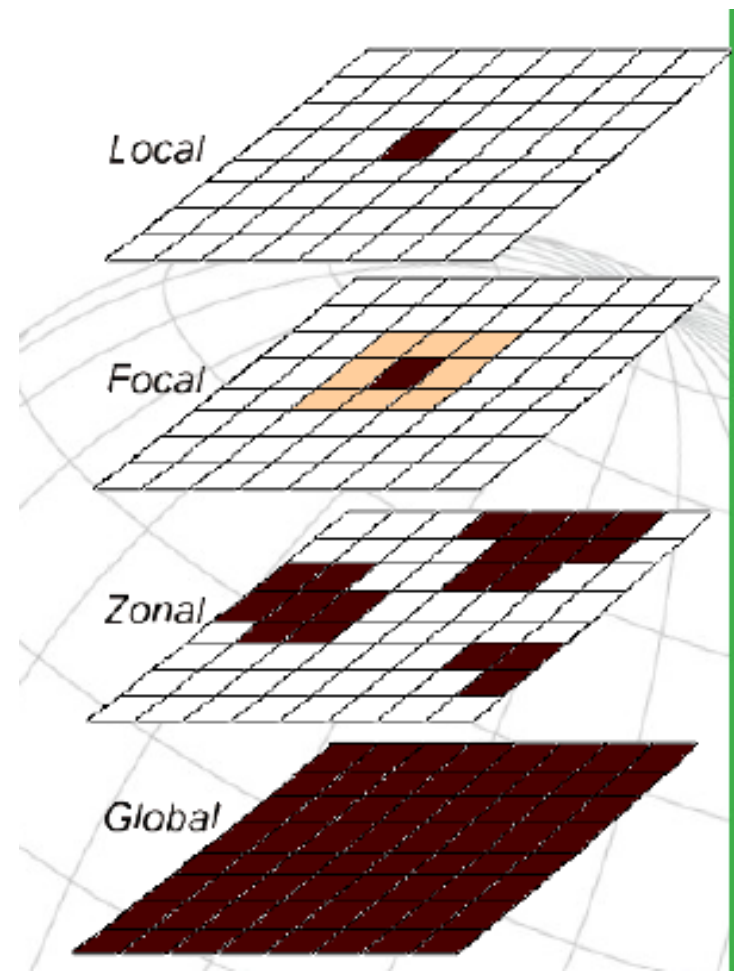


# Dělení funkcí mapové algebry

Z hlediska oblasti ze které je počítána hodnota výsledné buňky dělíme funkce mapové algebry na :

- **Lokální** - na individuální buňce, nová hodnota vzniká z individuální buňky jedné nebo více vrstev.
- **Fokální** - v definovaném okolí, nová hodnota vzniká z definovaného okolí buňky.
- **Zonální** - na specifické oblasti, nová hodnota vzniká ze zóny definované v jiné vrstvě.
- **Globální** - používají se všechny buňky informační vrstvy.

**Kartografické modelování**



# Lokální funkce

- **Lokální funkce** se obvykle dělí na:
- matematické, trigonometrické, exponenciální, logaritmické;
- reklasifikační;
- selekční;
- statistické.

# Reklasifikační funkce

- Mění hodnotu jednotlivých buněk na alternativní hodnoty pomocí různých metod.
  - Look up table
  - Reklasifikace pomocí individuálních hodnot
  - Reklasifikace pomocí tříd
  - Shlukování do intervalů či ploch

# Look up table

- Vytváří nový rastr pomocí vyhledávání hodnot v pomocné tabulce a definovaném sloupci.
  - $OutRas = Lookup(InRas1, "Category")$

0	1	1	0
3	3	1	2
■	0	0	2
3	2	1	0

InRas1

=

1	2	2	1
3	3	2	1
■	1	1	1
3	1	2	1

OutRas

■ Value = NoData

Value	Count	Code	Type	Category
0	5	10	PAX	Public
1	4	22	HAR	Private
2	3	14	WIN	Public
3	3	7	SAN	Federal

Value	Count	Category
1	8	Public
2	4	Private
3	3	Federal



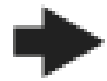
# Reklasifikace pomocí individuálních hodnot

- Mění hodnoty v poměru jedna k jedné na základě definovaných pravidel.

## Reclassification

3	3	19	1	6	6
20	3	19	17	1	5
20	15	15	6	11	14
12	7	15	8	8	10
13	4	18	18		10
16	4	18	7		9

Base Raster



Old Values	New Values
1-1	5
2-2	5
3-3	7
4-4	8
5-5	10
6-6	12
7-7	3
8-8	20
9-9	11
10-10	2
11-11	19
12-12	1
13-13	9
14-14	9
15-15	4
16-16	13
17-17	6
18-18	14
19-19	13
20-20	14



7	7	13	5	12	12
14	7	13	6	5	10
14	4	4	12	19	9
1	3	4	20	20	2
9	8	14	14		2
13	8	14	3		11

Output Raster

Value = NoData



# Reklasifikace pomocí tříd (range of values)

- Změna počtu či hodnoty tříd.
- Příklad:  $0 - 9 = 1$ ;  $20 - 30 = 5$ ;  $10 - 19 = 10$
- Pro souvislá data – není nutné definovat všechny hodnoty!
- Nástroj požaduje pouze určení spodní a horní hranice hodnot. Všechny hodnoty v rozmezí jsou reklasifikovány.
- Pozor na mezní hodnoty – potřeba se seznámit s pravidly konkrétního SW.

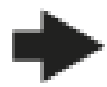




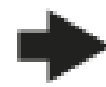
## Reclassification

3	3	19	1	6	6
20	3	19	17	1	5
20	15	15	6	11	14
12	7	15	8	8	10
13	4	18	18		10
16	4	18	7		9

Base Raster



Old Values	New Values
1-3	5
3-7	3
7-8	1
8-12	5
12-15	2
15-16	4
16-19	5
19-20	4
ND =	1



5	5	5	5	3	3
4	5	5	5	5	3
4	2	2	3	5	2
5	3	2	1	1	5
2	3	5	5	1	5
4	3	5	3	1	5

Output Raster

■ Value = NoData



# Shlukování do intervalů či ploch

- Rozmezí hodnot je reklasifikováno do:
  - stejných hodnotových intervalů,
  - stejných výsledných ploch
  - pomocí přirozených hranic (natural breaks).
- Nástroj Slice v ArcGIS

3	3	19	1	6	6
20	3	19	17	1	5
20	15	15	6	11	14
12	7	15	8	8	10
13	4	18	18		10
16	4	18	7		9

**Base Raster**



**Slice  
10  
Interval**



**Reclassification**

2	2	10	1	3	3
10	2	10	9	1	3
10	8	8	3	6	7
6	4	8	4	4	5
7	2	9	9		5
8	2	9	4		5

**Output Raster**

☐ Value = NoData

3	3	19	1	6	6
20	3	19	17	1	5
20	15	15	6	11	14
12	7	15	8	8	10
13	4	18	18		10
16	4	18	7		9

**Base Raster**



**Slice  
5 Equal  
Areas**



**Reclassification**

1	1	5	1	2	2
5	1	5	4	1	2
5	4	4	2	3	4
3	2	4	2	2	3
3	1	4	4		3
4	1	4	2		3

**Output Raster**

☐ Value = NoData

# Lokální funkce ArcGIS

- Lokální statistické funkce
- Kombinace více vstupních rastrů (Combine)
- **Nalezení počtu výskytů** splňujících určitá kritéria - *Equal To Frequency, Greater Than Frequency a Less Than Frequency.*
- **Nalezení hodnoty** splňující určitá kritéria - *Popularity a Rank.*
- **Nalezení polohy** splňující určitá kritéria.



# Lokální statistické funkce

- Vypočítá pro jednotlivé buňky statistiku z daného počtu rastrů.
- Majority, Maximum, Mean, Median, Minimum, Minority, Range, Standard Deviation, Sum a Variety.
- **Mean - průměr**
- Vypočítá průměr z jednotlivých buněk vstupních rastrů.
- Výsledek má vždy hodnotu „floating point.“
- `OutRas = CellStatistics(["InRas1", "InRas2", "InRas3"], "Mean")`

1	1	0	0
	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
	0	0	2
3	2	1	0

InRas2

	1	0	0
2	0	3	3
0	0	3	2
1	1		0

InRas3

=

	1.0	0.33	0.0
	1.3	2.0	2.3
	0.0	1.0	2.0
2.7	1.0		0.3

OutRas



# Lokální statistické funkce

- **Majority** – nejčastěji se vyskytující hodnota v jednotlivých buňkách.
- **Integer, floating point.**
- **Pokud je více možných výsledků, pak NoData.**
- $OutRas = CellStatistics([InRas1, InRas2, InRas3], "Majority")$

1	1	0	0
	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
	0	0	2
3	2	1	0

InRas2

	1	0	0
2	0	3	3
0	0	3	2
1	1		0

InRas3

=

	1	0	0
			2
	0	0	2
			0

OutRas



# Lokální statistické funkce

- **Maximum (median, minimum, minority, range, standard deviation, sum)**
- Nejvyšší hodnota (stejně tak pro ostatní statistické funkce)
- Integer, floating point – input=output
- `OutRas = CellStatistics(["InRas1", "InRas2", "InRas3"], "Maximum")`

1	1	0	0
	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
	0	0	2
3	2	1	0

InRas2

	1	0	0
2	0	3	3
0	0	3	2
1	1		0

InRas3

=

	1	1	0
	3	3	3
	0	3	2
4	2		1

OutRas

# Lokální statistické funkce

- **Variety – variabilita**
- Určí počet unikátních hodnot v jednotlivých vstupech a buňkách.
- Výstupní rastr je vždy integer.
- $OutRas = CellStatistics([InRas1, InRas2, InRas3], "Variety")$

1	1	0	0
0	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
0	0	0	2
3	2	1	0

InRas2

0	1	0	0
2	0	3	3
0	0	3	2
1	1	0	0

InRas3

=

0	1	2	1
0	3	3	2
0	1	2	1
3	3	0	2

OutRas



# Kombinace více rastrů

- Funkce **Combine** kombinuje několik vstupních rastrů a přiřadí novou hodnotu všem unikátním kombinacím jednotlivých buněk. Původní hodnoty jednotlivých rastrů jsou zapsány do atributové tabulky výstupního rastru.
- Každá unikátní kombinace je označena novou hodnotou.
- Názvy vstupních rastrů jsou užity jako označení sloupců nové atributové tabulky a označují rodičovství nově vzniklých atributů.

# Combine

1	1	0	0
■	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
■	0	0	2
3	2	1	0

InRas2

=

1	2	3	4
■	5	6	7
■	4	4	7
8	9	2	1

OutRas

■ Value = NoData

Value	Count	Code
0	5	002
1	5	004
2	3	005
4	2	008

Value	Count	Type
0	5	PAX
1	4	HAR
2	3	WIN
3	3	SAN

Value	Count	InRas1	InRas2
1	2	1	0
2	2	1	1
3	1	0	1
4	3	0	0
5	1	1	3
6	1	2	1
7	2	2	2
8	1	4	3
9	1	0	2

$OutRas = Combine([InRas1, InRas2])$

# Počet výskytů splňujících určitá kritéria

- **Less than Frequency**

- Nepovinný rastr může specifikovat počet výskytů.

- Výstupní rastr je vždy integer

- $OutRas = LessThanFrequency(ValRas, [InRas1, InRas2, InRas3])$

2	2	2	2
2	2	2	2
2	2	2	2
2	2	2	2

ValRas

1	1	0	0
1	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
0	0	0	2
3	2	1	0

InRas2

1	1	0	0
2	0	3	3
0	0	3	2
1	1	0	0

InRas3

=

3	3	3
2	1	0
3	2	0
1	2	3

OutRas



# Hodnoty splňující určitá kritéria

**Popularity** – pořadí n-tých výskytů – pozor na striktní pravidla počítání (NoData, všechny odlišné hodnoty).

**OutRas = Popularity(ValRas, [InRas1, InRas2, InRas3])**

2	2	2	2
2	2	2	2
2	2	2	2
2	2	2	2

ValRas

1	1	0	0
	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
	0	0	2
3	2	1	0

InRas2

	1	0	0
2	0	3	3
0	0	3	2
1	1		0

InRas3

=

	1	1	0
			3
	0	3	2
			1

OutRas



# Hodnoty splňující určitá kritéria

- Rank Hodnoty vstupních rastrů jsou seřazeny podle buněk, výstupní hodnota pořadí je určena pomocným rastrem.
- $OutRas = Rank(ConstRas, [InRas1, InRas2, InRas3])$

3	3	3	3
3	3	3	3
3	3	3	3
3	3	3	3

ConstRas

1	1	0	0
■	1	2	2
4	0	0	2
4	0	1	1

InRas1

0	1	1	0
3	3	1	2
■	0	0	2
3	2	1	0

InRas2

■	1	0	0
2	0	3	3
0	0	3	2
1	1	■	0

InRas3

=

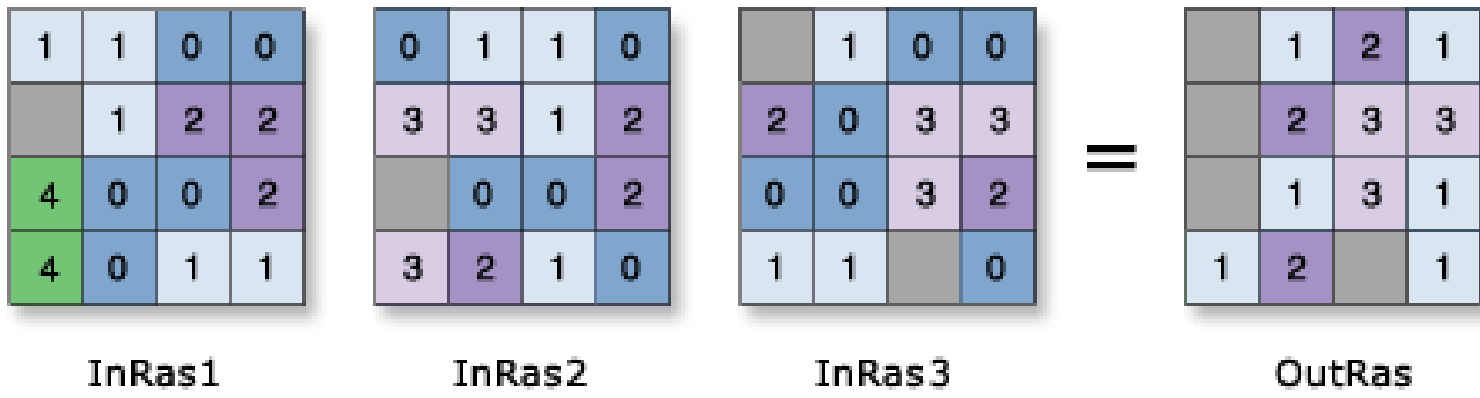
■	1	1	0
■	3	3	3
■	0	3	2
4	2	■	1

OutRas

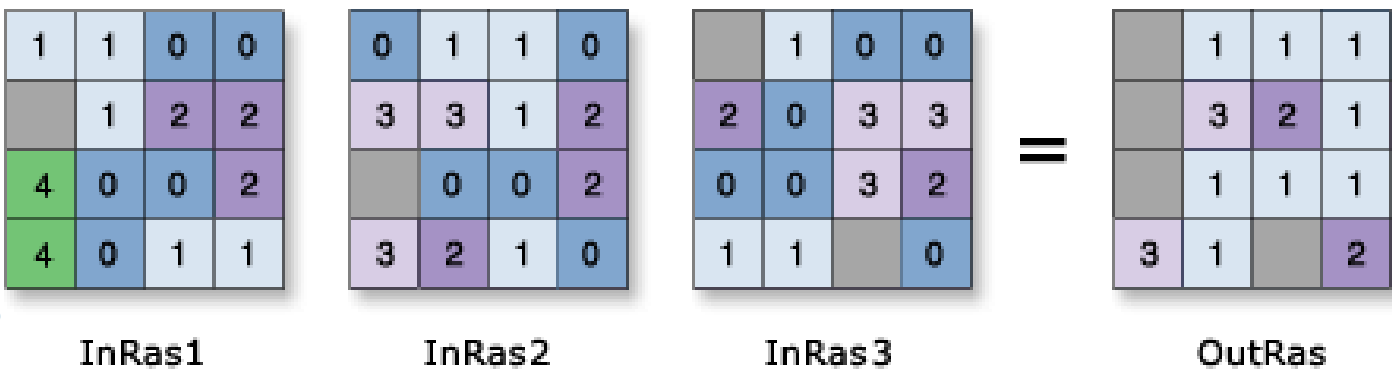


# Nalezení polohy splňující určitá kritéria.

- ***OutRas = HighestPosition([InRas1, InRas2, InRas3])***



- ***OutRas = LowestPosition([InRas1, InRas2, InRas3])***





# Výběr pomocí podmínky – conditional

- Kontroluje výsledné hodnoty na základě podmínek, které jsou aplikovány na vstupní hodnoty.
- Podmínky lze uplatnit na atributy či polohu buněk.
- Dotaz (podmínka) na atributy explicitně identifikuje všechny buňky, které jsou hodnoceny jako „True“.
- True buňkám jsou následně přiřazeny nové hodnoty (případně NoData).
- False buňkám jsou přiřazeny hodnoty podle podmínky.
- **Nástroje Con, Pick**

- **Provede podmíněčný výběr na základě požadavku a splnění podmínky.**
- $OutRas = Con(InRas1, 40, 30, "Value \geq 2")$


1	1	0	0
NoData	1	2	2
4	0	0	2
4	0	1	1

InRas1

=

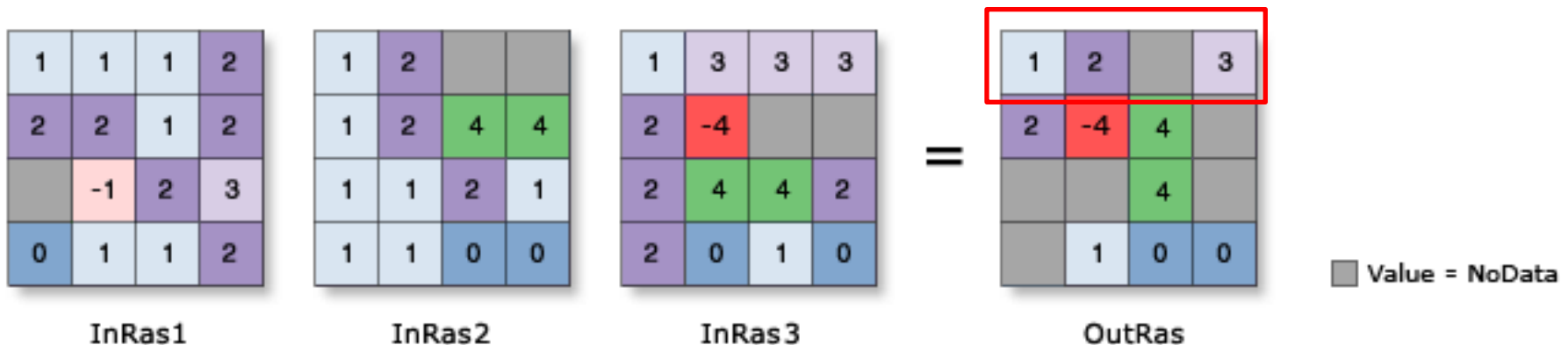
30	30	30	30
NoData	30	40	40
40	30	30	40
40	30	30	30

InRas2

 Value = NoData



- Hodnota z pozičního rastru je použita k určení toho, z jakého vstupního rastru má být použita hodnota pro výstupní rastr.
- $OutRas = Pick(InRas1, [InRas2, InRas3])$



# No Data

- Pokud je hodnota buňky definována jako prázdná (NoData), znamená to, že tato **buňka nenes**e žádnou informaci o **prostoru**, který reprezentuje. Zacházení s tímto druhem hodnot se podstatně liší. Buňky s prázdnou hodnotou mohou být zpracovány dvojím způsobem:
  - Přiřazením **prázdné hodnoty buňce výstupního rastru**, pokud existuje prázdná hodnota této polohy v jakémkoli vstupním rastru. V tomto případě se to týká vstupních rastrů zpracovaných lokálními funkcemi. V případě fokálních funkcí se prázdná hodnota objeví v místech, kde se v okolí zpracovávané buňky vyskytuje prázdná hodnota. V případě zonálních funkcí by se jednalo o zónu.
  - Druhou možností je **ignorování prázdné hodnoty** a provedení výpočtu pouze s existujícími hodnotami. Tato možnost neсе určité riziko, protože výstupní hodnoty nesou určitou nepřesnost plynoucí z toho faktu, že nemáme žádné informace o buňce nesoucí prázdnou hodnotu.
- 0 je validní hodnota
- 999 obvykle použito pro No data