

Geografické informační systémy

Analýza rastrových dat

Úvod

- ◆ Papírová mapa je pro počítač nulovou informací. Po zpracování mapové předlohy do tématických vrstev se z ní stává část GIS databáze.
- ◆ Potřeba data:
 - ◆ Uložit
 - ◆ Zpracovat
- ◆ GIS umožňuje analýzy = odvozování nových znalostí z již existujících (uložených).
 - ◆ velmi expertní činnost.
- ◆ Experimentování s modelem.
- ◆ Analyticky orientované GIS aplikace.

Oblasti analýz

- ◆ Prohledávání databáze
- ◆ Rastry (analýzy povrchů):
 - ◆ Mapová algebra, statistiky
 - ◆ Vzdálenostní analýzy
 - ◆ Analýza modelů terénu
- ◆ Vektory (analýzy topologie):
 - ◆ Analýza sítí – spojení, zóny dostupnosti.
- ◆ Analýza obrazů (z DPZ) – například tvorba ortofotomap, klasifikace snímků z DPZ, ...
- ◆ Analytické operace (postupy) – automatizované experimenty

Dotazy na databázi

◆ Prohledávání databáze.

- ◆ data se do IS ukládají pro potřeby uživatelů
- ◆ data uložená, data odvozená (analýzou)
- ◆ typicky: "vyber z DB údaje, které vyhovují podmínce P a proved' na nich operaci O"

◆ V GIS:

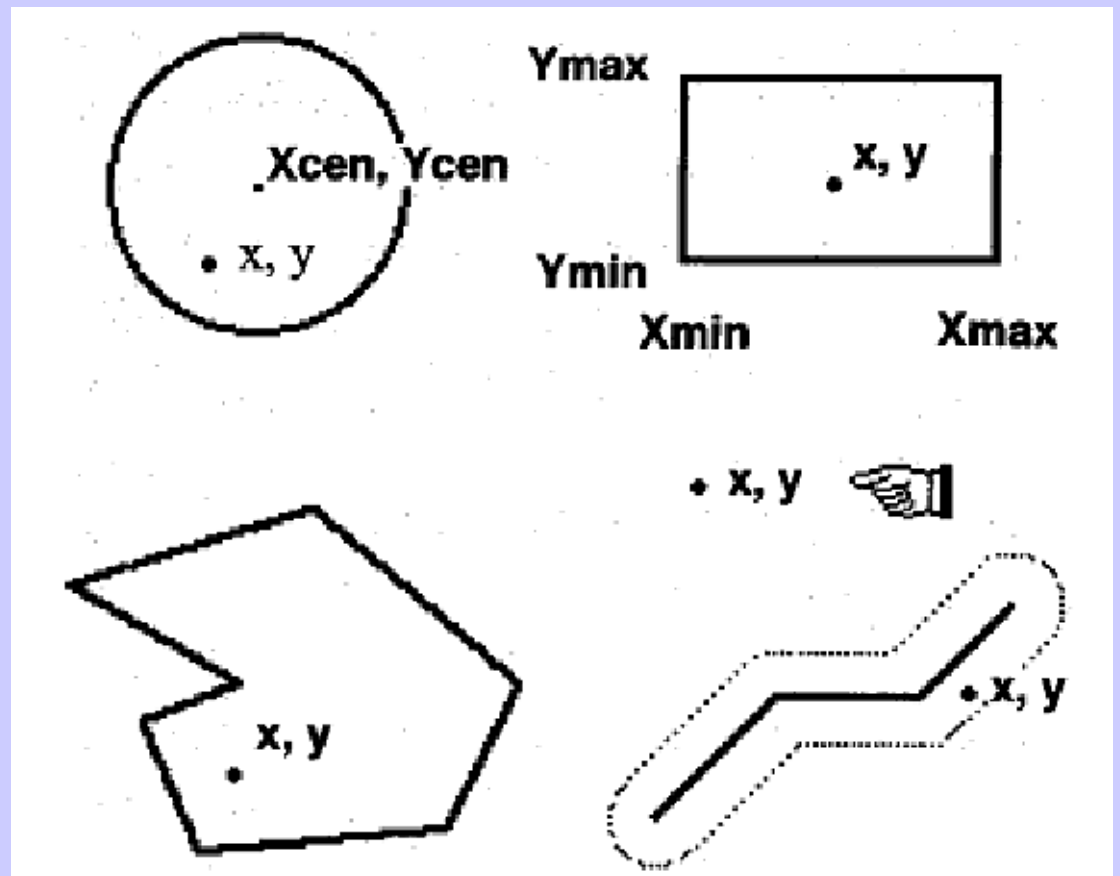
- ◆ *prostorový dotaz* - "co je na daném místě", "objekty ve vzdálenosti od místa" – d.what.rast, d.what.vect, buffer, ...
- ◆ *atributový dotaz* - "které objekty mají danou vlastnost" – selekce (d.vect obce where....)
- ◆ komplikované (kombinované) dotazy - "zobraz zamokřené plochy s výměrou větší než 1 ha, které sousedí s průmyslově využívanými plochami"

Prostorové dotazy

- ◆ příkazy d.what.vect a d.what.rast v GRASSu
- ◆ zadání souřadnice na příkazovém řádku, myší
- ◆ objekt (vektor) se identifikuje a zobrazí své atributy
- ◆ přesnost zadání souřadnice, okolí, analýza vektorové vrstvy
- ◆ v rastru se identifikuje konkrétní buňka (souřadnicí-do které plošky souřadnice padne, udáním pozice v matici rastru)
 - ◆ resampling

Prostorové dotazy

- ◆ Problém identifikace vektorového objektu lze převést na problém "padne bod do určité plochy"



Atributové dotazy

- ◆ Vyhledání objektu podle jeho identifikátoru (cat, jméno) – "navez='Svratka'"
- ◆ častěji však podle podmínky na jeho atributy:
- ◆ "select * from obce where ob01>50000"
- ◆ ... vyber z parcel ty, které jsou volné...
- ◆ Selekce pro účely:
 - ◆ provedení analytické operace
 - ◆ zjednodušení zobrazení (zejména pak v mapových výstupech)

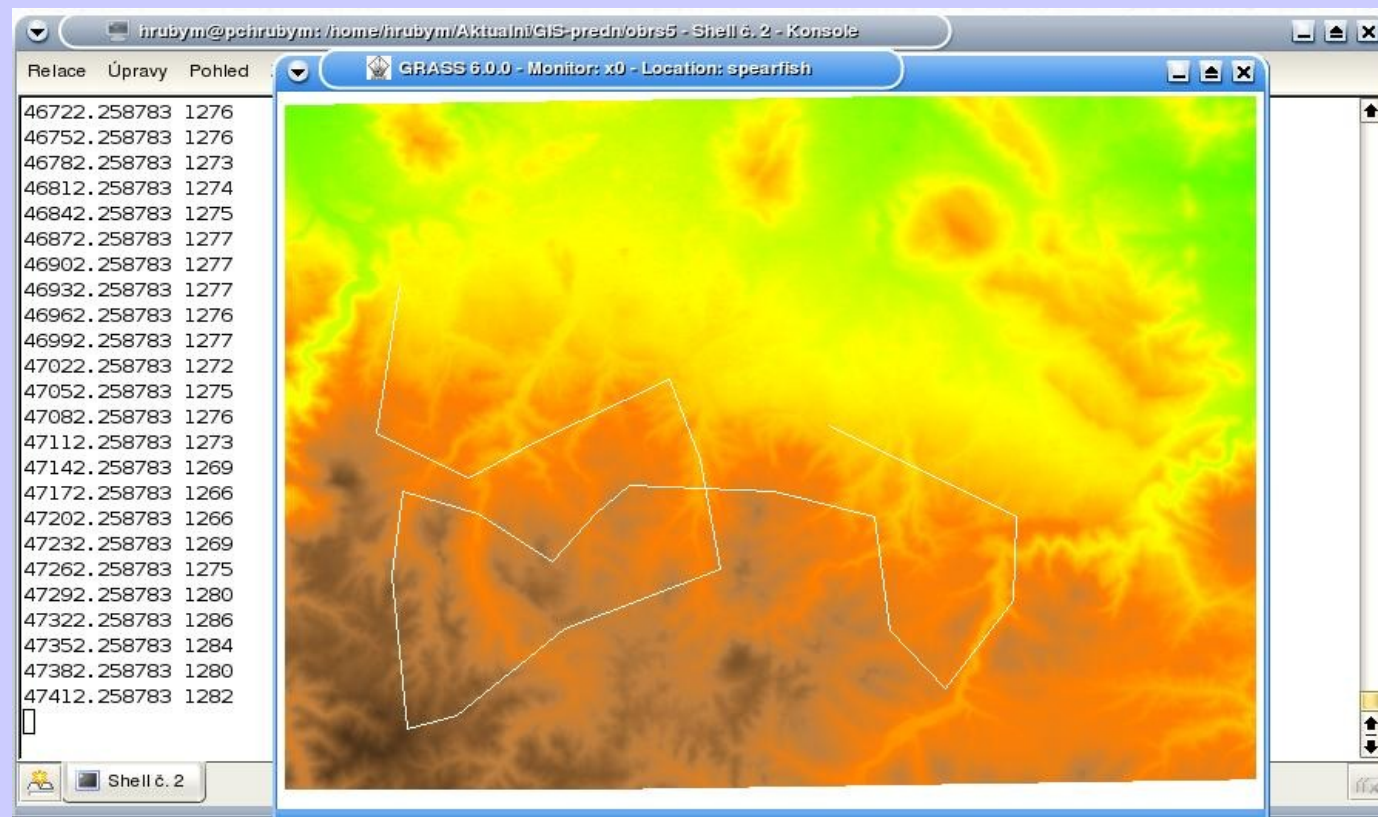
- ◆ v GRASSu:
 - ◆ db.columns - výpis názvů všech sloupců
 - ◆ db.execute - vykonání SQL příkazu
 - ◆ db.select - vykonání příkazu "select" (parametrem je table, where)
 - ◆ v.db.connect - napojení tabulky na vektorovou vrstvu
- ◆ d.vect obce disp=attr attrcol=nazob where="ob01>50000"
- ◆ db-select se provádí nad tabulkou atributů, pak se vykreslí objekty (cat) podle vybraných řádků

Atributové dotazy v rastru

- ◆ chceme vidět (uložit) pouze buňky mající atribut, který vyhovuje dané podmínce
 - ◆ `d.rast dem_srtm vallist=100-500`
 - ◆ `d.rast geology catlist=4,5`
- ◆ reklasifikace (`r.reclass`) – selekce s následným uložením (pohledu)
- ◆ maska (0,1): zobrazení rastu přes masku
 - ◆ GRASS: přejmenovat libovolnou mapu na MASK
- ◆ Vykonání kombinovaného dotazu nad více rastry.

r.profile

- ◆ Příklad prostorového dotazu v rastru.
- ◆ Aplikování dotazu na buňky ležící na zadané trase
- ◆ Profil cesty přes DEM
- ◆ Vzorek dat po nějaké linii



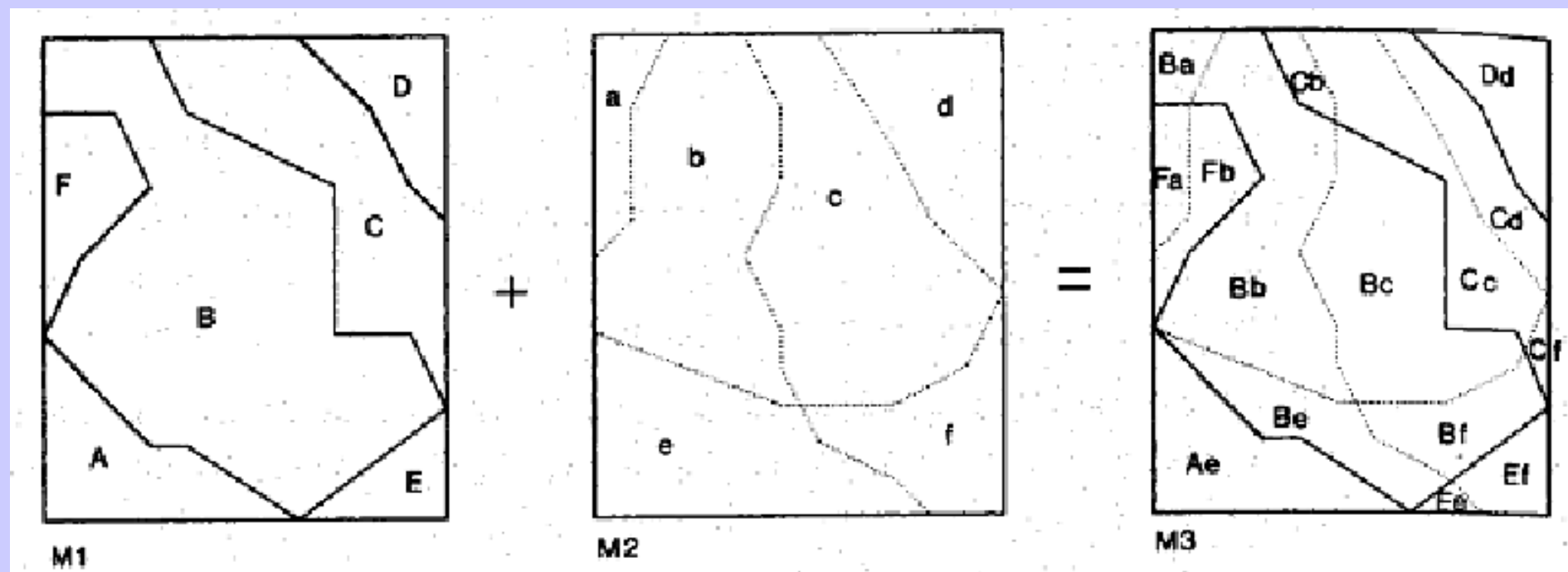
Dotazy na topologické údaje

- ◆ „jaké je ohraničení polygonu" - hraniční linie
- ◆ „na jaké linie navazuje....“
- ◆ směr linie
- ◆ s jakým polygonem sousedí daný polygon přes danou linii (část ohraničení)
- ◆ Snahou vektorových formátů je tyto informace mít předvypočtené/uložené.
 - ◆ problém při editaci geometrie
 - ◆ obvykle se topologie přepočítá na explicitní žádost uživatele

Překrytí – analýza vektorů

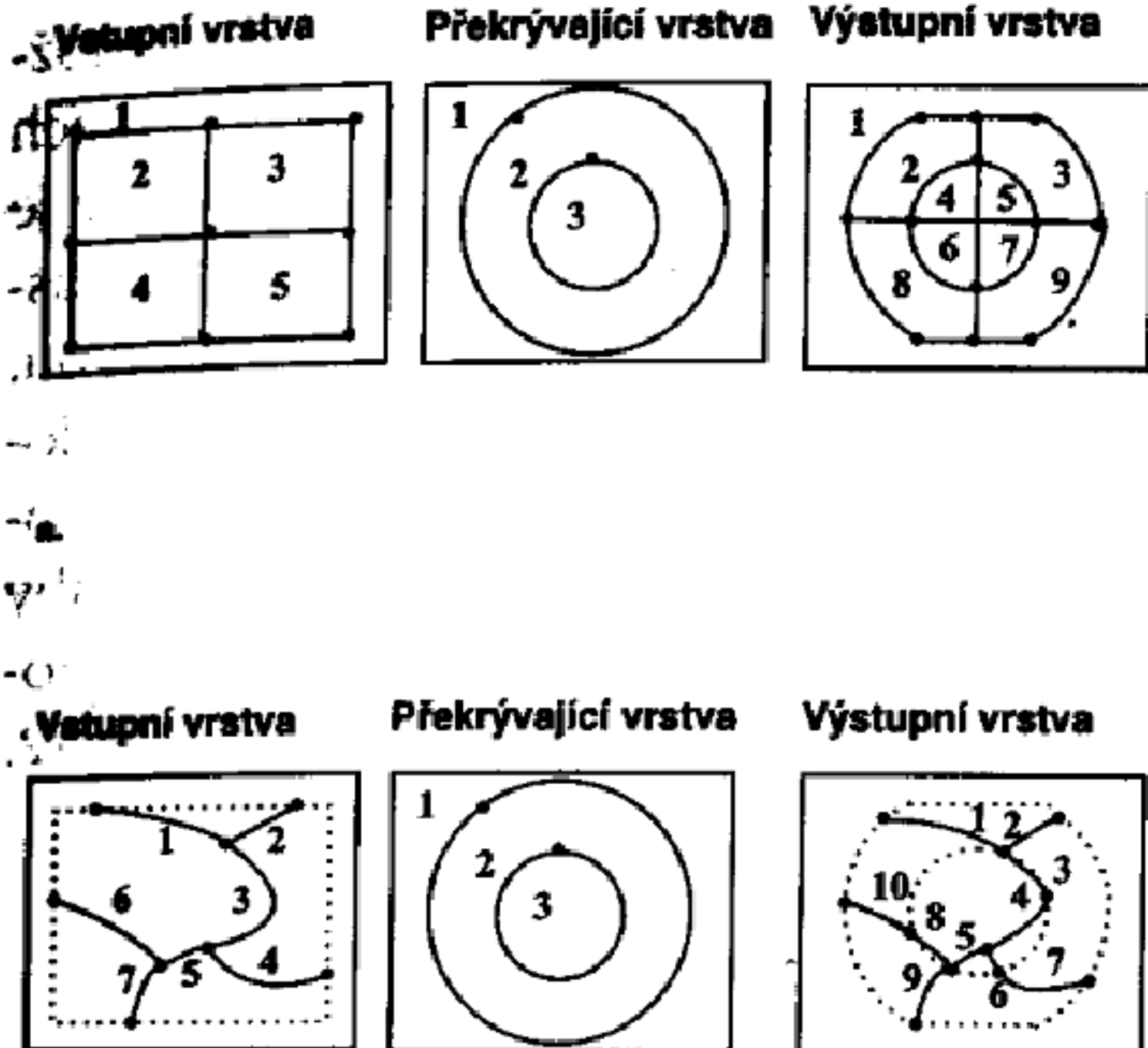
- ◆ Velmi častý typ vektorových analýz.
Analýza nad více vektorovými vrstvami.
- ◆ klasický postup: průhledné fólie, více map
- ◆ překrytí polygonů, linií
- ◆ vytvoření záznamu o nově vzniklých objektech, kombinace atributů
- ◆ hledání průsečíků linií, identifikace nových polygonů=clipping

Topologické překrytí dvou map



Topologické překrytí dvou map

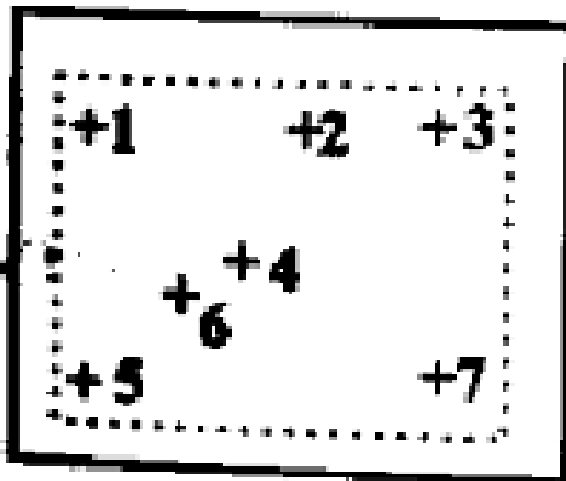
- ◆ polygon v polygonu
- ◆ čára v polygonu



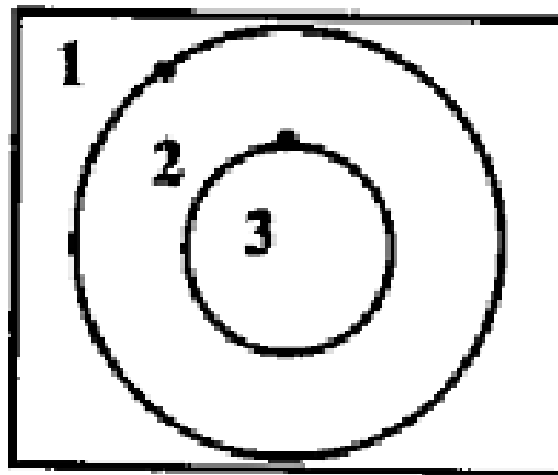
Topologické překrytí dvou map

- ◆ bod v polygonu

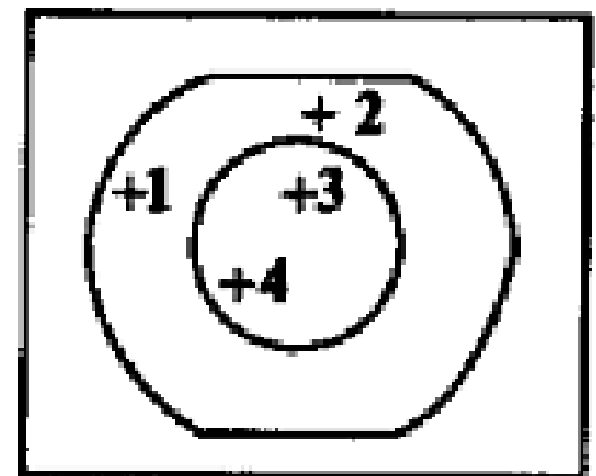
Vstupní vrstva



Překrývající vrstva



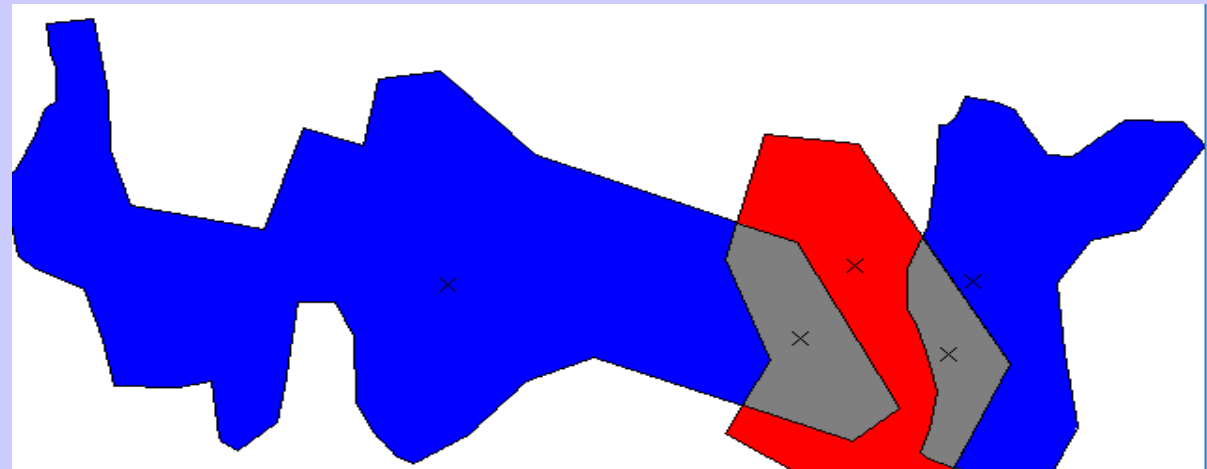
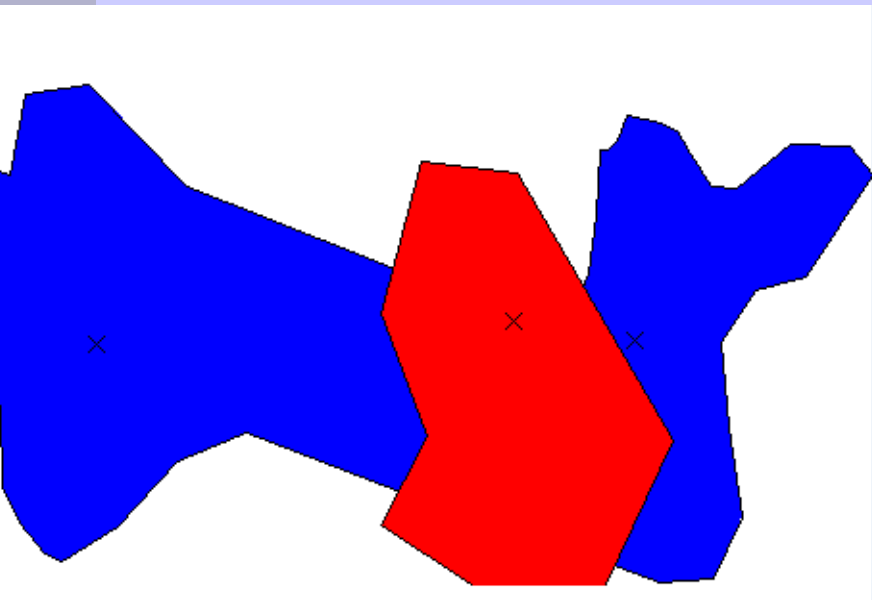
Výstupní vrstva



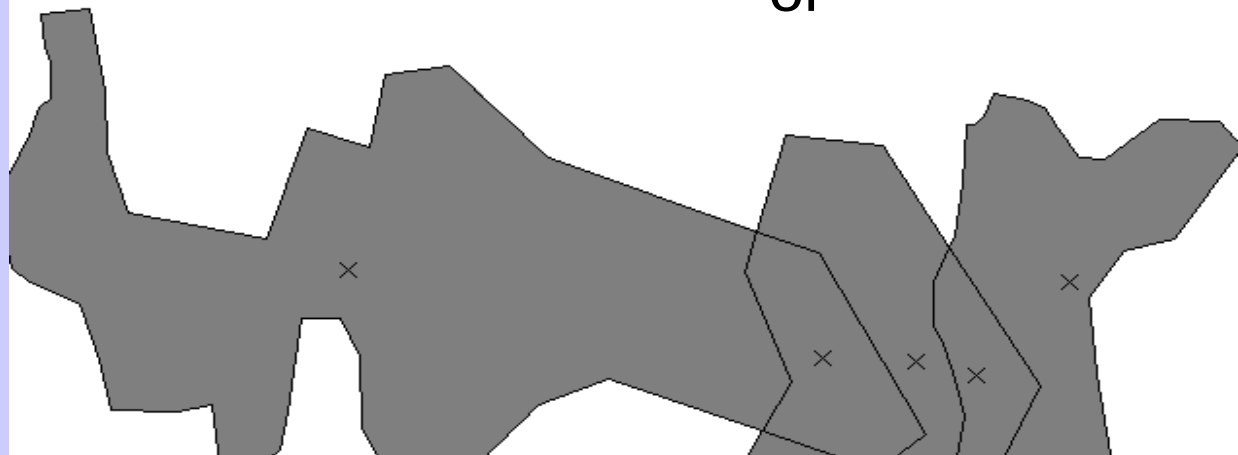
Operace nad polygony

- ◆ průnik, sjednocení
- ◆ řešení vztahů bod, linie, polygon v polygonu
- ◆ v.overlay ainput=pl1 binput=pl2 output=prun2
operator=and

and



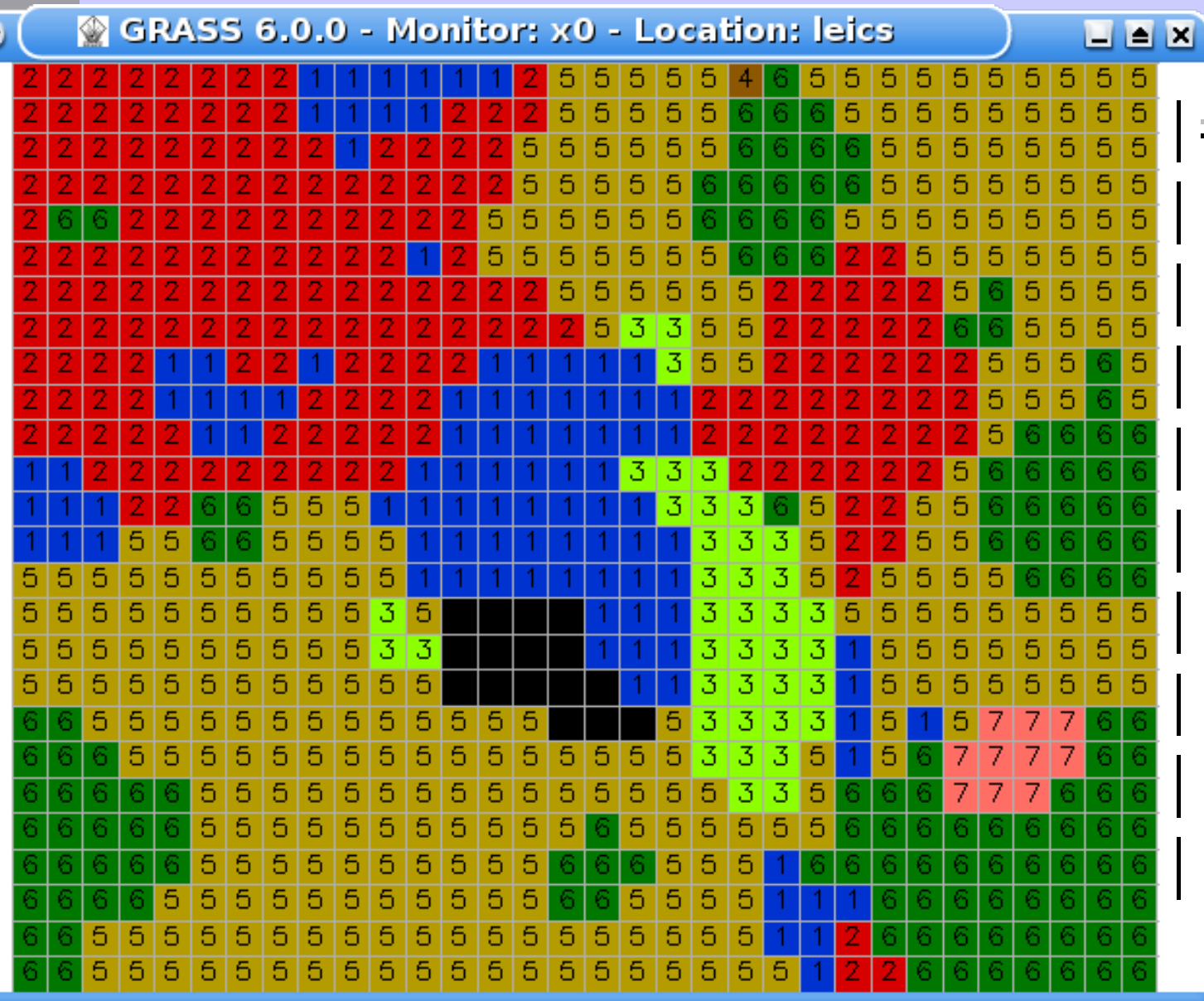
or



Překrytí rastrů

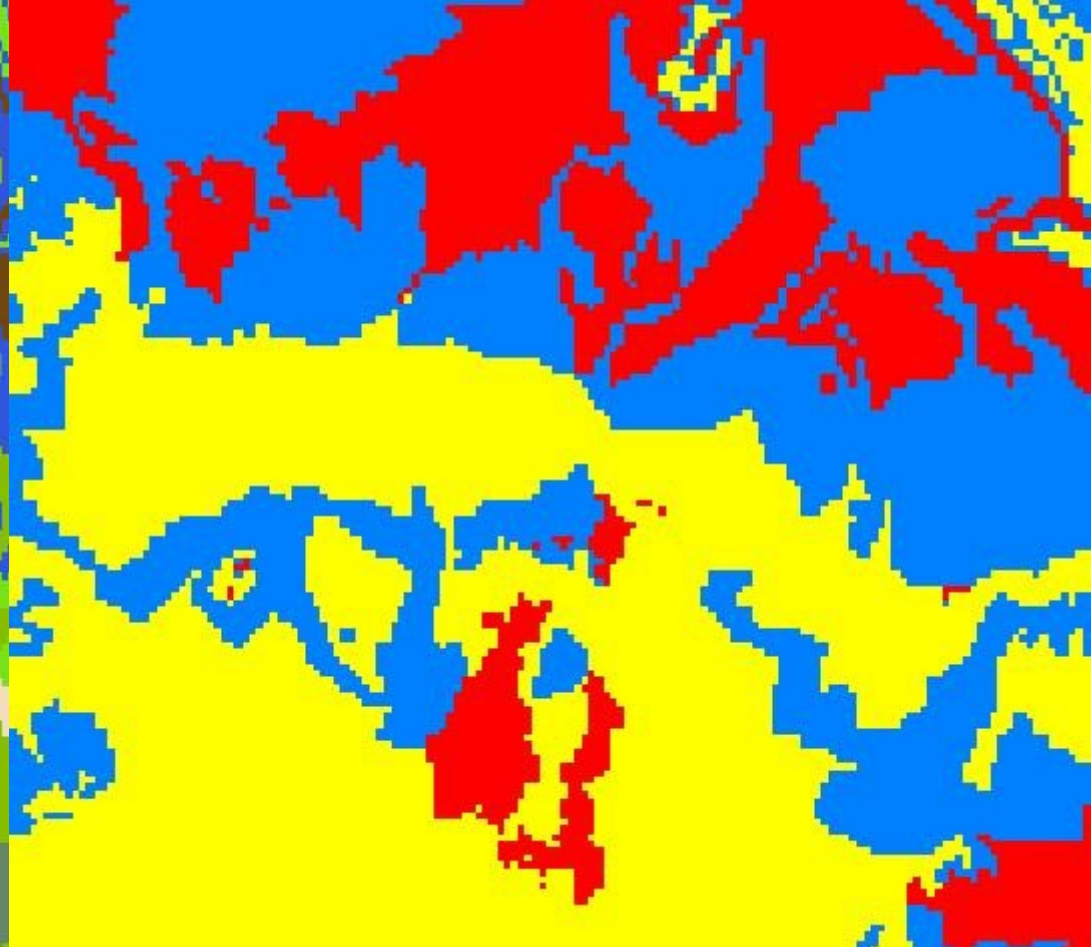
- ◆ Dvě základní operace:
 - ◆ Reklasifikace - `r.reclass`
 - ◆ mapová algebra - `r.mapcalc`
- ◆ Poznámka: grafické překrytí - nemá nic společného s analýzou (můžeme považovat pouze za „náhled“)
 - ◆ Grafické překrytí s vektory
 - ◆ `d.rast mapa1`
 - ◆ `d.rast -o mapa2`
 - ◆ `d.rast -o mapa3`

Rastrová analýza



r.reclass – rastrová reklasifikace

- ◆ Mapování atributů vstupního rastru do atributů výstupního
- ◆ `r.reclass input= output=`
- ◆ Reklasifikační tabulka:
 - ◆ `L = R [text]`
 - ◆ `A thru B = R`
 - ◆ `* = R`
 - ◆ `end`
 - ◆ Implicitně: `* = NULL`
- ◆ GIS systémy se liší pouze v podobě předání reklasifikační tabulky.



Mapová algebra

- ◆ Metody matematického kombinování map.
- ◆ `r.mapcalc`
- ◆ vyhradně na rastrových mapách
- ◆ definice: mapová algebra je nástroj, který dovoluje zpracovávat rastrové reprezentace reality s použitím jazyka mapové algebry
- ◆ operátory, funkce

Mapová algebra - operátory

- ◆ aritmetické (+, -, ...)
- ◆ booleovské (true, false)
- ◆ relační (<, >, =)
- ◆ bitové posuvy, kombinatorické, logické, akumulativní, přiřazovací

Mapová algebra - funkce

- ◆ lokální - nad hodnotou jedné buňky - kombinace hodnot ve více vrstvách
- ◆ fokální - operace nad okolím dané buňky
- ◆ zonální - vykonávají se nad specifické oblasti informační vrstvy
 - ◆ statistické operace
 - ◆ geometrické op. - například zjištění obvodu zóny
- ◆ globální - týkají se všech buněk vrstvy (analýzy frikčních povrchů)
- ◆ Operace - nad jednou vrstvou, více vrstvami

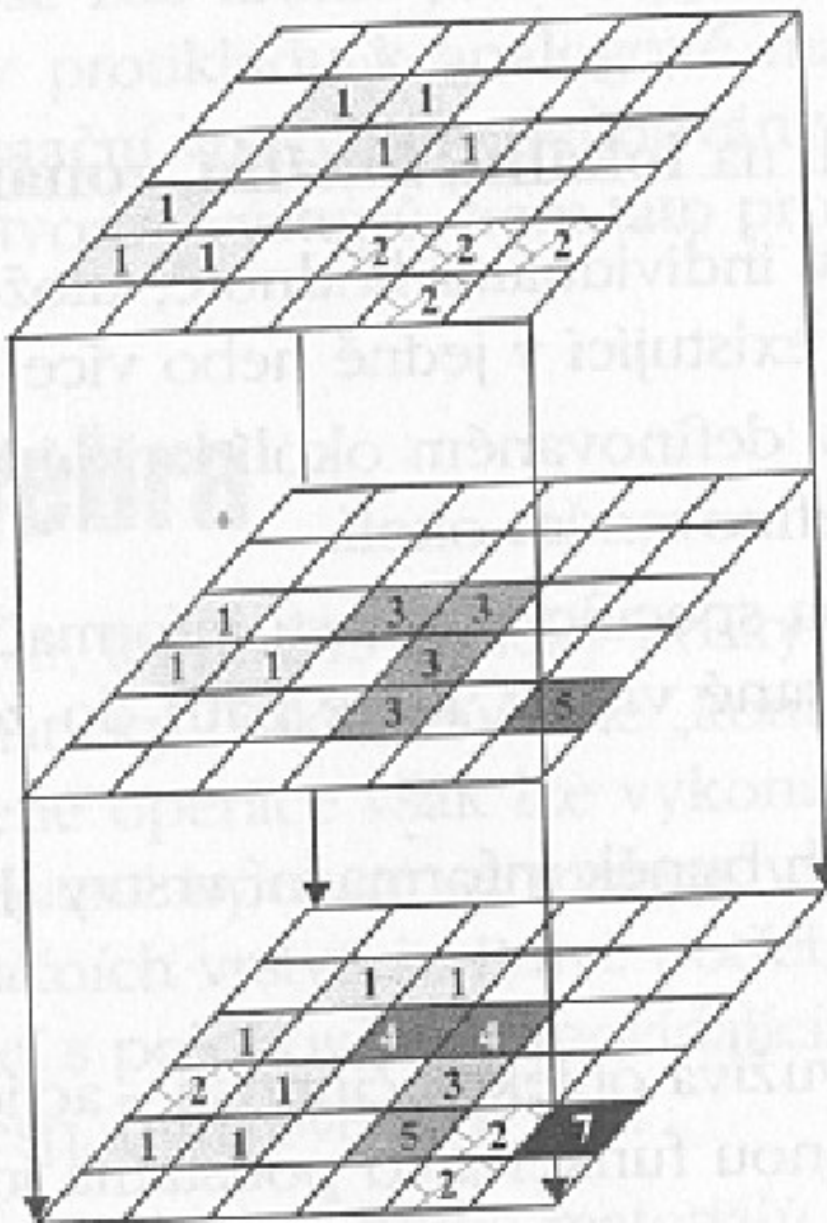
Mapová algebra - lokální fce

Vstupní obraz č. 1,
„INGRID 1“

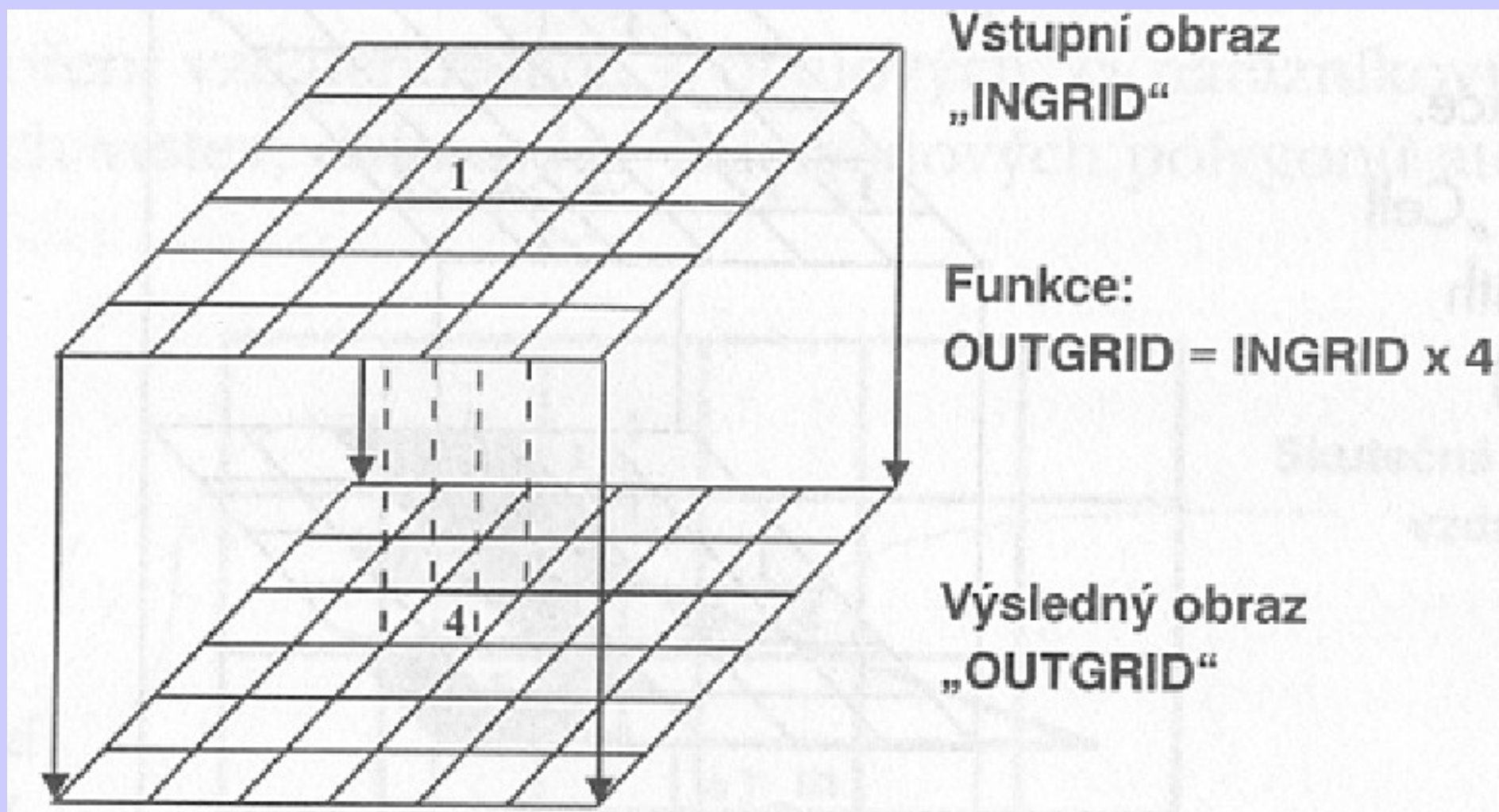
Vstupní obraz č. 2,
„INGRID 2“

Funkce:
 $\text{OUTGRID} = \text{INGRID 1} + \text{INGRID 2}$

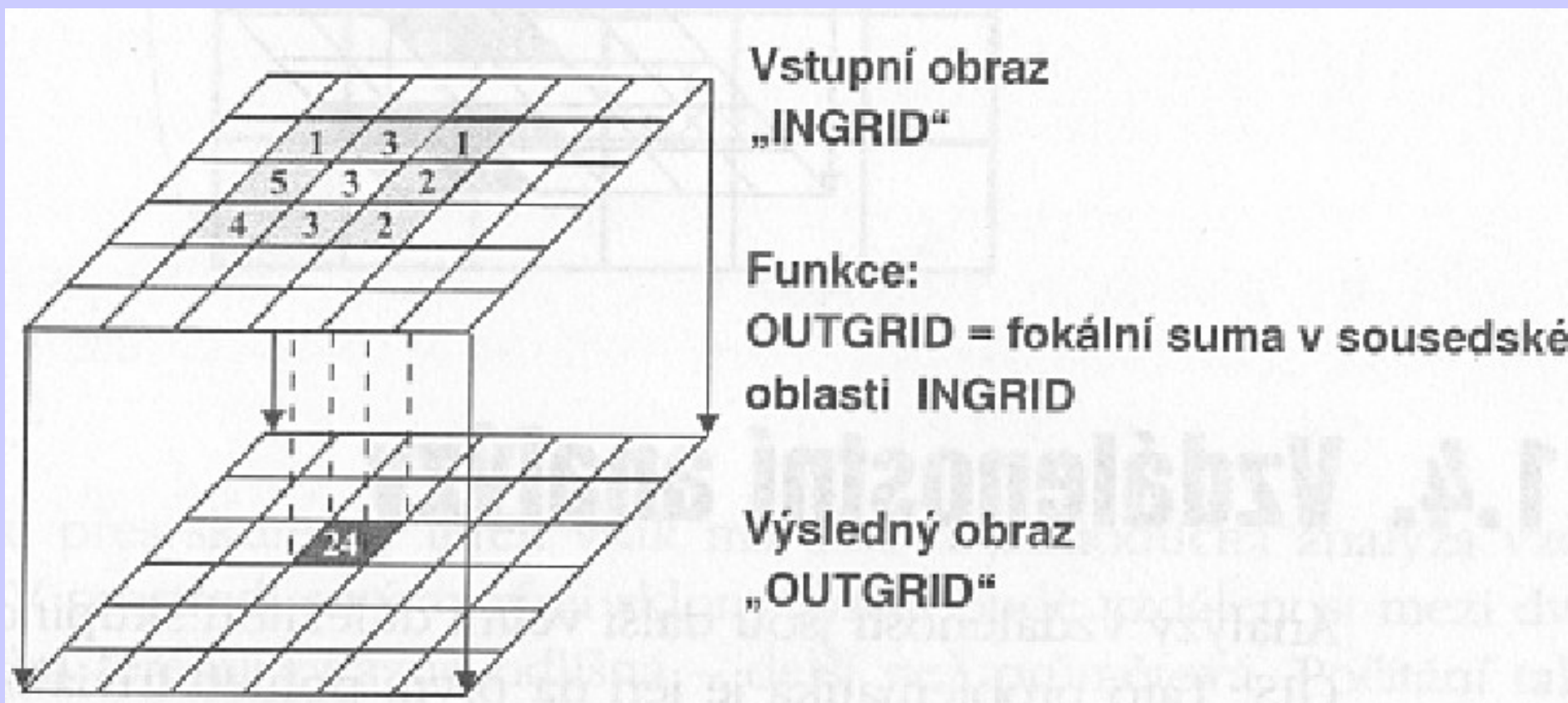
Výsledný obraz
„OUTGRID“



Mapová algebra - lokální fce



Mapová algebra - fokální fce



GRASS - r.mapcalc

- ◆ přičtení rastrové mapy "výšky budov" s výškovým modelem terénu
 - ◆ $dembudovy = budovy + dem$
- ◆ Normalizovaný vegetační index:
 - ◆ $NDVI = (TM4 - TM3) / (TM4 + TM3)$
- ◆ Uložení reklasifikované mapy (pohledu):
 - ◆ `echo „4 9 = 1“ | r.reclass input=geology output=pisky`
 - ◆ `echo „nova=pisky“ | r.mapcalc`

r.mapcalc

- ◆ konstrukce if(podminka, then_value, else_value)
- ◆ pracuje lokálně
- ◆ nova = if(mapa>10, mapa, 0)
- ◆ negace = if(mapa=1,0,1)
- ◆ Fokální funkce:
 - ◆ res = inp[0,0] + inp[-1,-1]*3 + ...

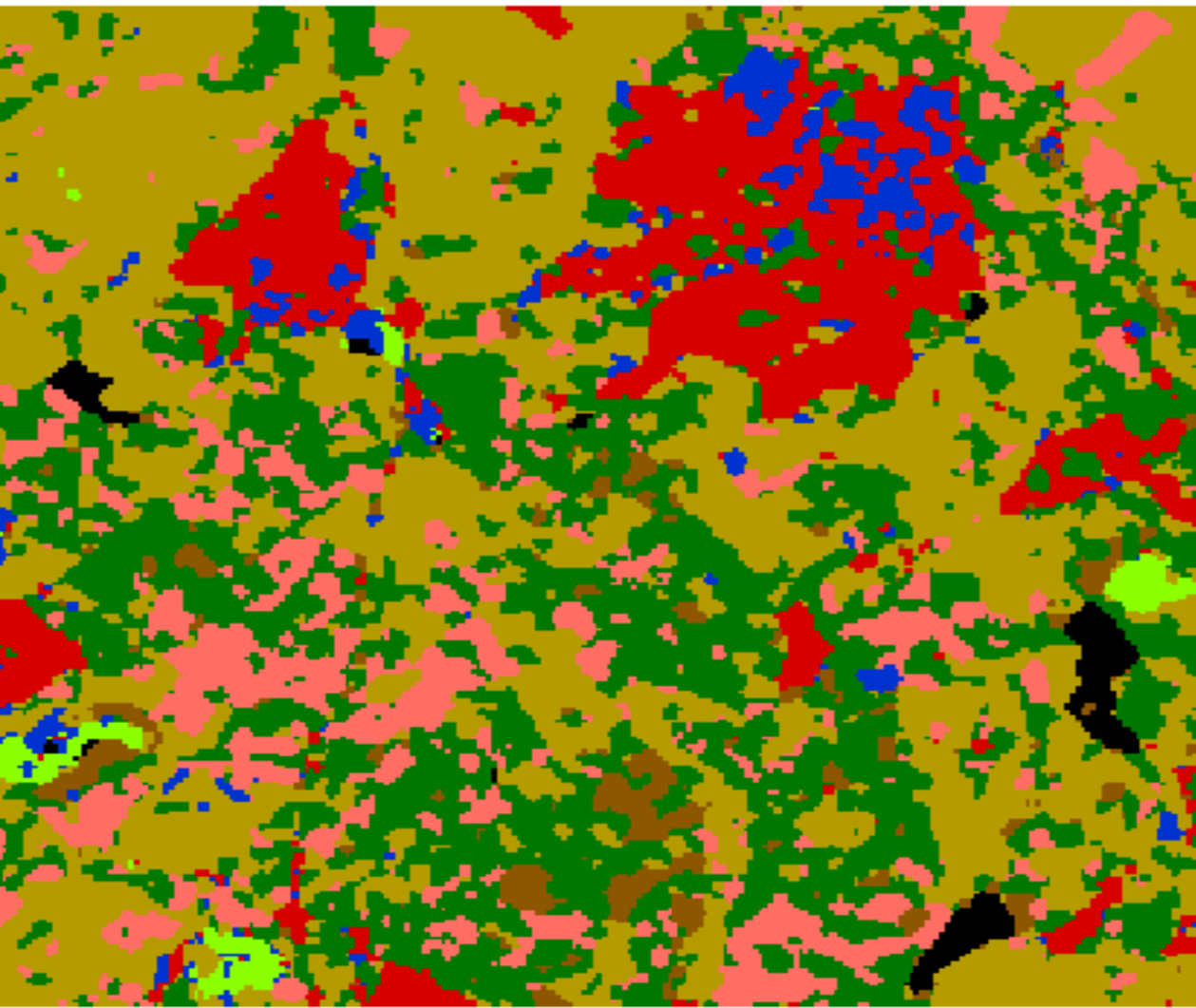
r.mfilter

- ◆ Aplikace filtru na rastr *r.mfilter* (matice koeficientů), dělitel
- ◆ *r.neighbors* – určení buňky jako výsledku operace nad okolím buňky
 - ◆ Průměr, mode (nejčastější), minimum, maximum
- ◆ Zpracování obrazu – typicky z DPZ

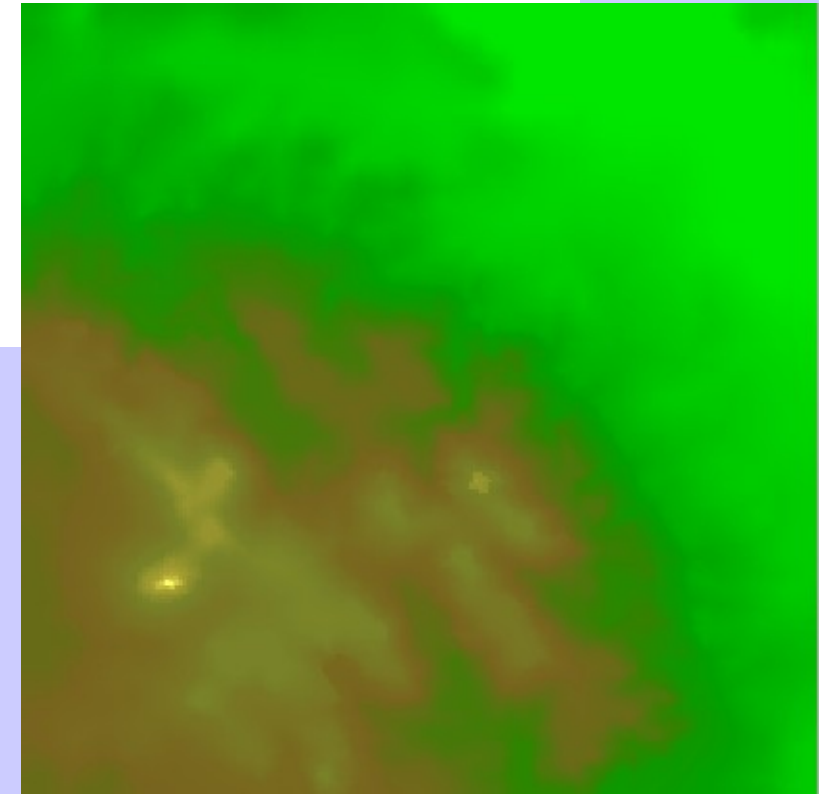
Příklad

- ◆ Mapa "landcov" využití (bydlení, průmysl, voda) povrchu.
- ◆ Mapa "topo" výšek.
- ◆
- ◆ `r.average base=landcov cover=topo
output=avheight`
- ◆ "avheight" obsahuje průměrnou výšku spočtenou na jednotlivých kategoriích využití.

landcov



- | | |
|---|----------------|
|  | 1) Industry |
|  | 2) Residential |
|  | 3) Quarry |
|  | 4) Woodland |
|  | 5) Arable |
|  | 6) Pasture |
|  | 7) Scrub |
|  | 8) Water |



Mapy: landcov, topo

Comments:

recode of raster file landcov

0:0:0.0000000000

1:1:83.6110113704

2:2:76.3006634499

3:3:135.2540540541

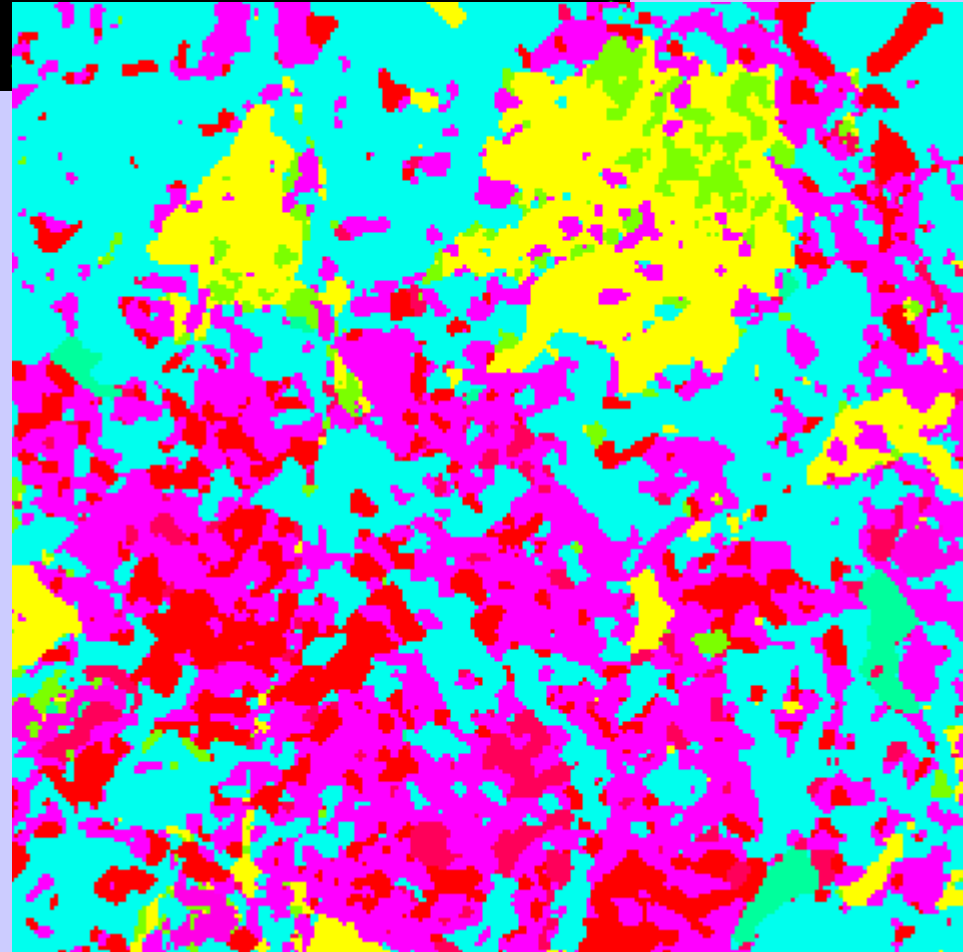
4:4:142.9445397370

5:5:104.0214288832

6:6:133.4833491912

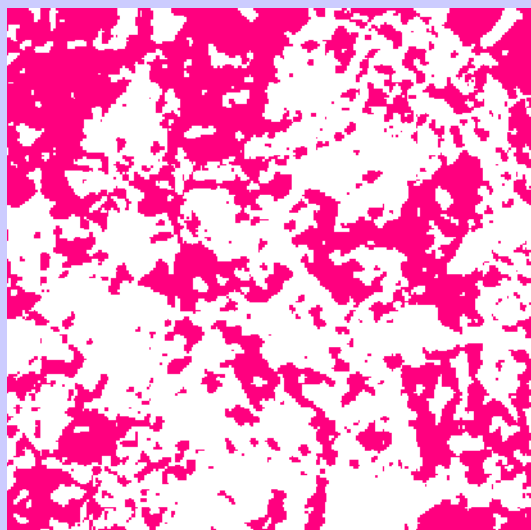
7:7:148.2487038731

8:8:99.2691218130



Příklad

- ◆ Sečtení délky řek na obdělávaných a průmyslových územích.
- ◆ reklasifikace landcov - vybrat políčka arable a industry,
- ◆
- ◆ echo "1 5 = 1" | r.reclass input=landcov output=vv
- ◆ r.mapcalc, vvy=vv*water (průnik)
- ◆ r.report vvy units=cell
- ◆ g.remove rast=vv,vvy



MASK:none		

MAP: (untitled) (vvy in prednaska)		

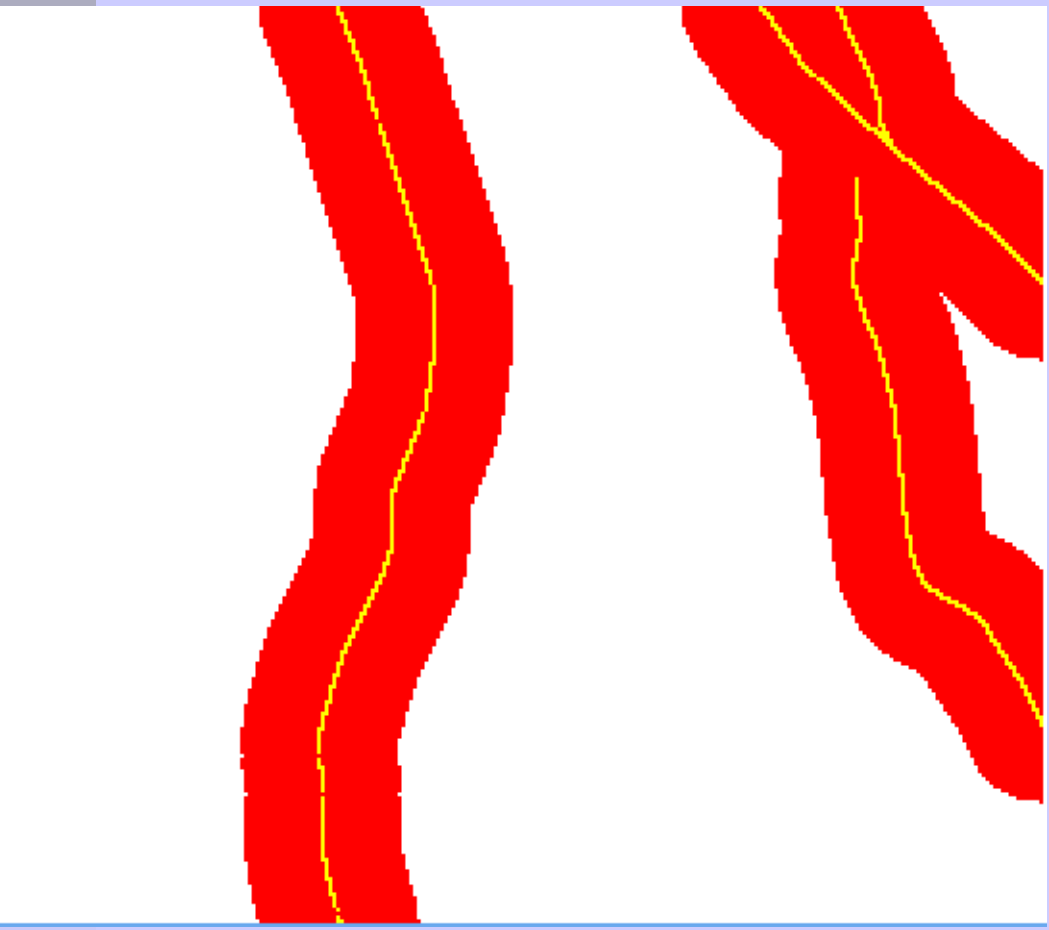
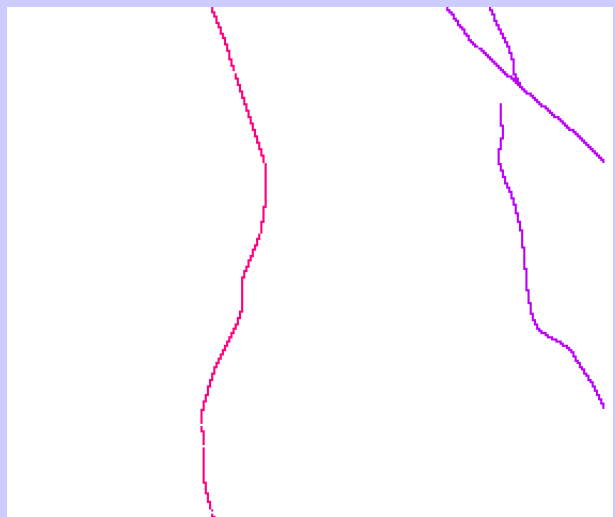
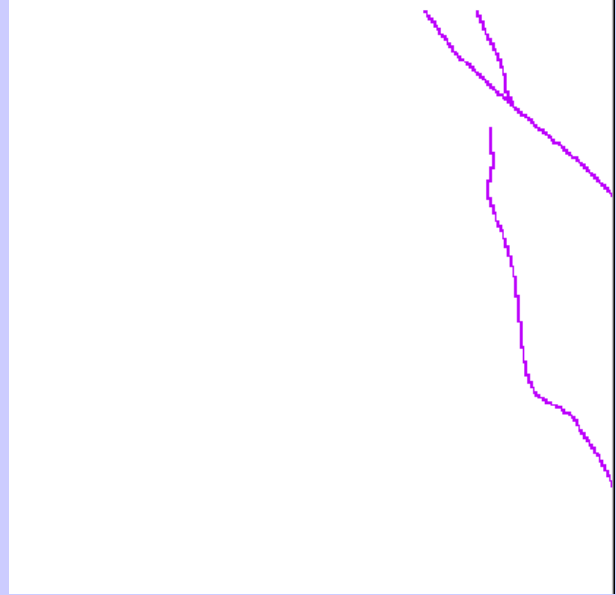
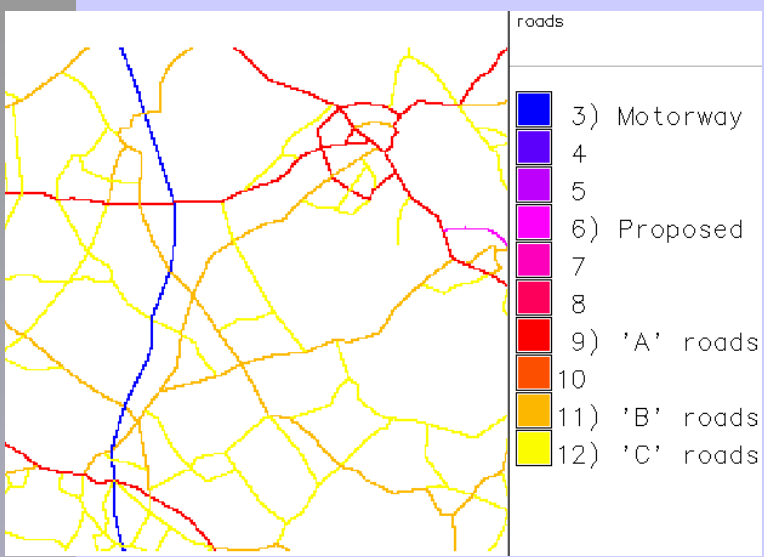
Category Information		cell
# description		count

4	.	508
* no data.	.	57092

TOTAL		57600
+-----+		

Příklad – zapojení bufferingu

- ◆ Zobrazte obydlená místa v dosahu 1km od M1 a železnice.
- ◆ reklasifikace roads, rail
- ◆ sloučení do jedné mapy - `r.patch input=in1,in2 output=in`
- ◆ `r.buffer input=in output=inbuffer distances=1000 units=meters` `//r.grow`
- ◆ reklasifikace inbuffer - `inin`
- ◆ `r.mapcalc: (jinak lze g.rename rast=inin,MASK)`
- ◆ `prisk = inin * popln`

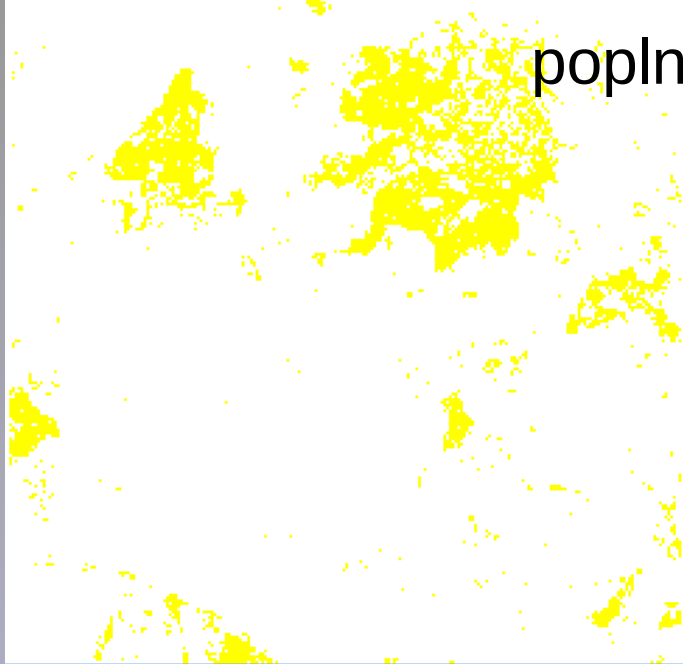


MAP: Distance Zones (inbuffer in prednaska)	

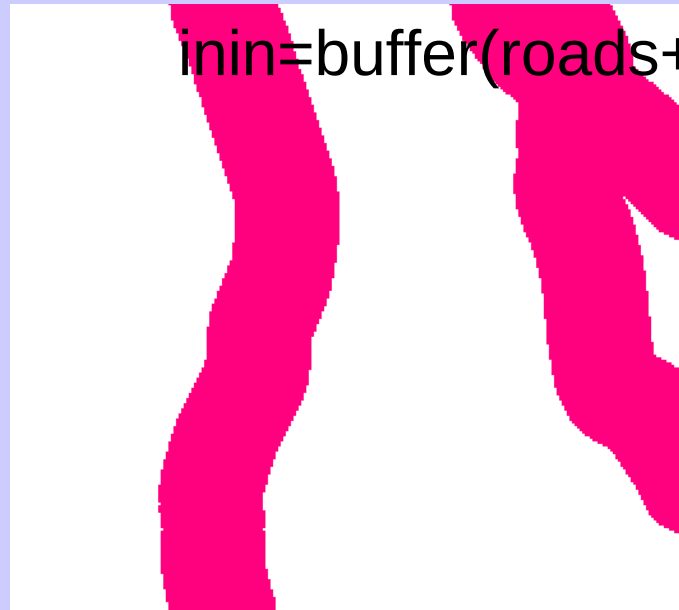
	Category Information
#	description

1	distances calculated from these locations
2	0-1000 meters
*	no data
+-----	

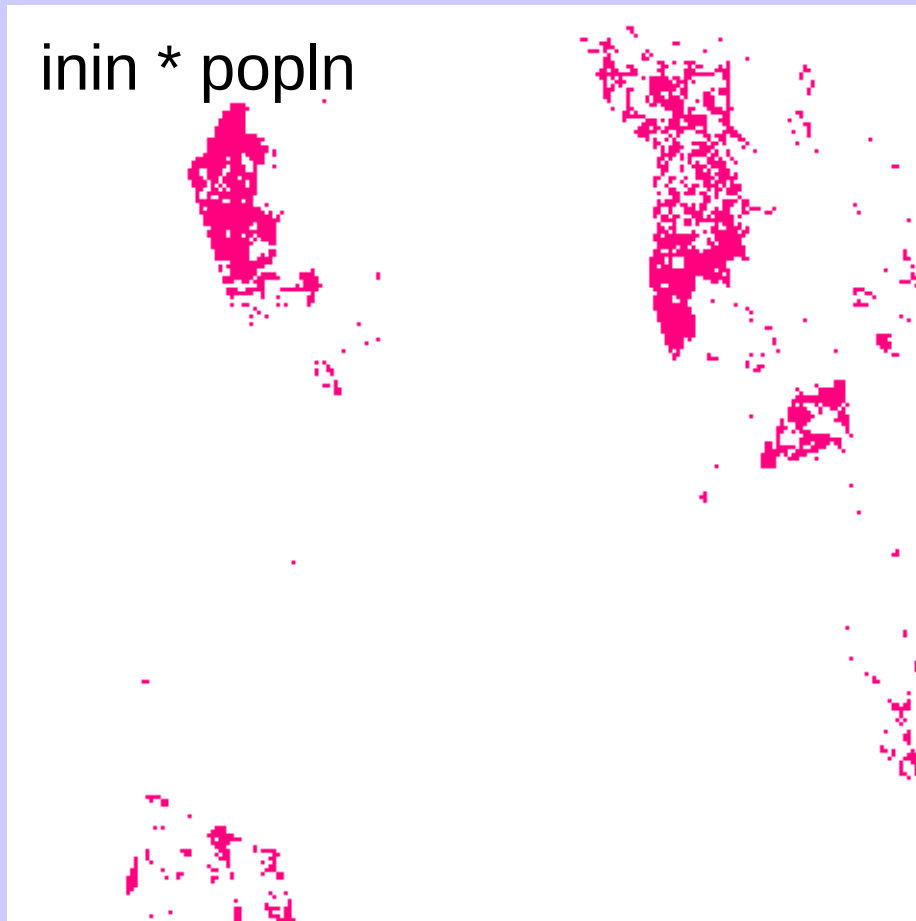
popln



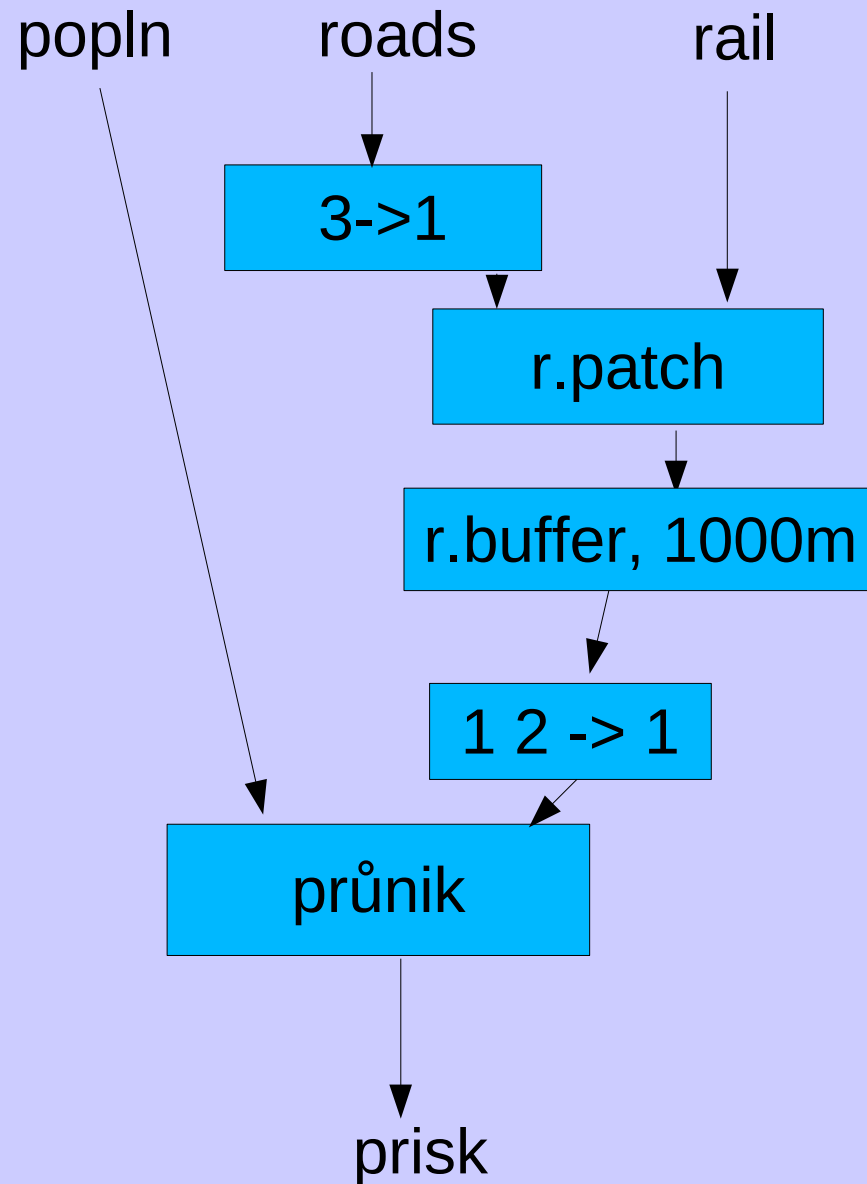
inin=buffer(roads+rail)



inin * popln



Model operace

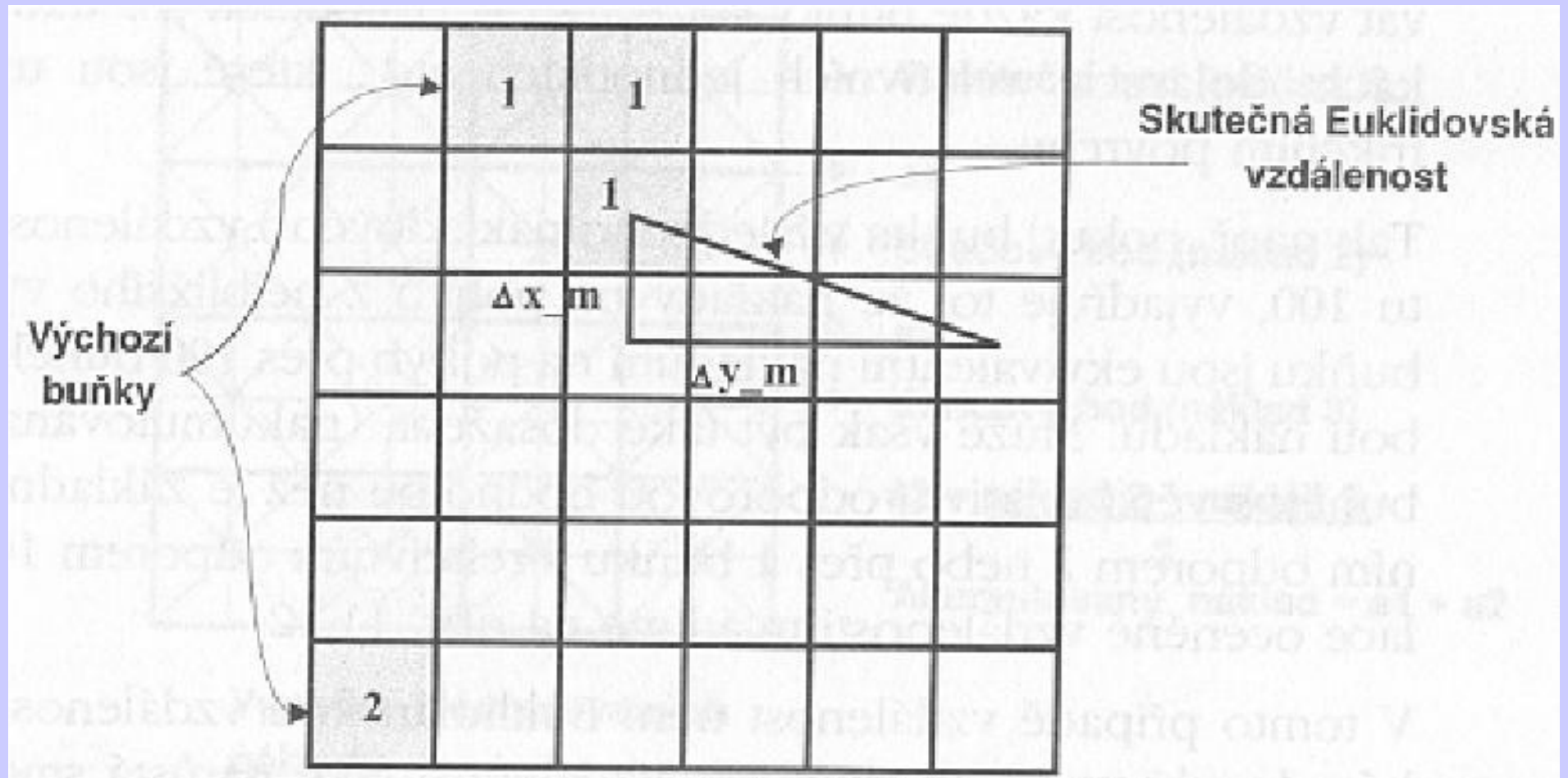


Vzdálenostní analýzy

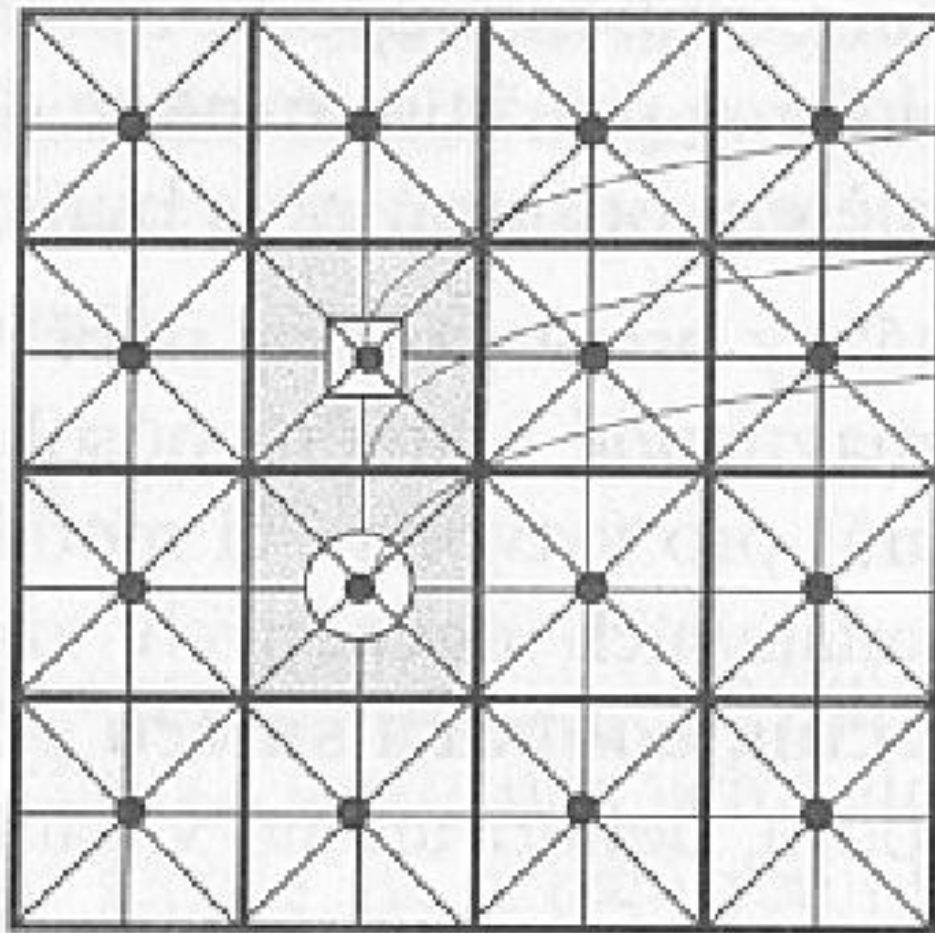
- ◆ klasická analýza v GIS, není až tak triviální
- ◆ euklidiánská vzdálenost
- ◆ reálné (nebo dokonce horské) prostředí - vzdálenost není jenom "vzdušná"
- ◆ nákladová vzdálenost (náklady na pohyb po krajině)
- ◆ nákladový vzdálenostní povrch (NVP)
- ◆ rastrový NVP, kde každá buňka obsahuje "nákladovost" pro pohyb přes její plochu = frikční povrch

- ♦ význam - hledání cesty s nejnižším nákladem
- ♦ vztaženo k výchozímu bodu - kumulativní náklad (každá buňka vyjadřuje svou hodnotou sumu nákladů...)

Vzdálenost



Akumulované nákladové vzdálenosti



Počáteční bod (náklad 1)

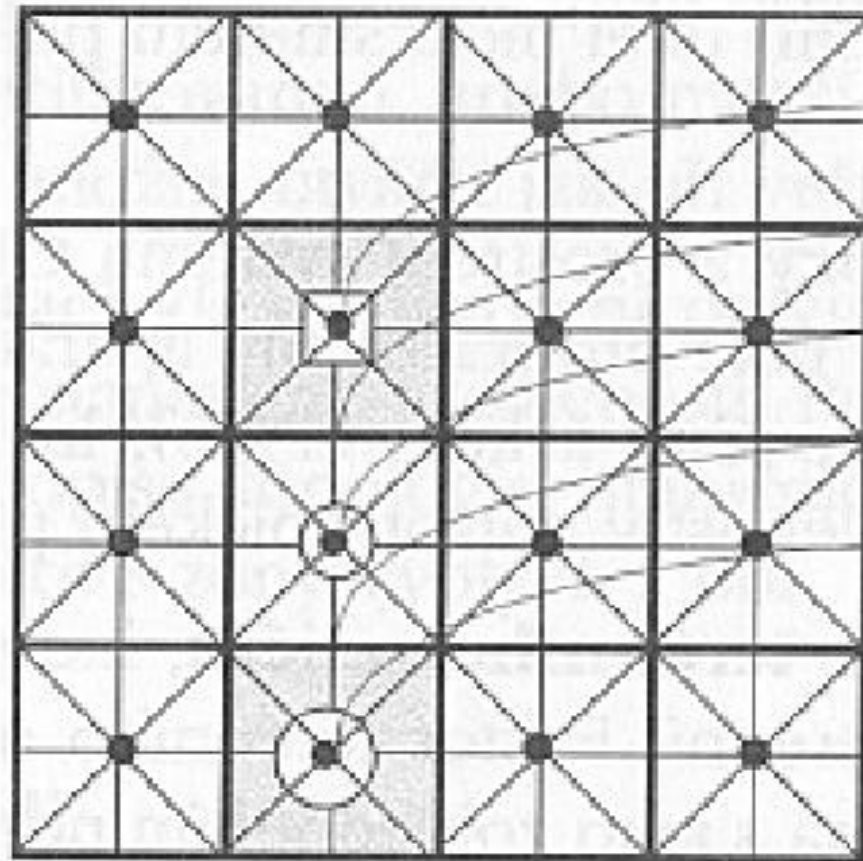
a1

Koncový bod (náklad 2)

$$a1 = \frac{\text{náklad 1} + \text{náklad 2}}{2}$$

Výpočet nákladů
v horizontálním
a vertikálním směru

Akumulované nákladové vzdálenosti



► Počáteční bod (náklad 1)

► a_1

► Středový bod (náklad 2)

► a_2

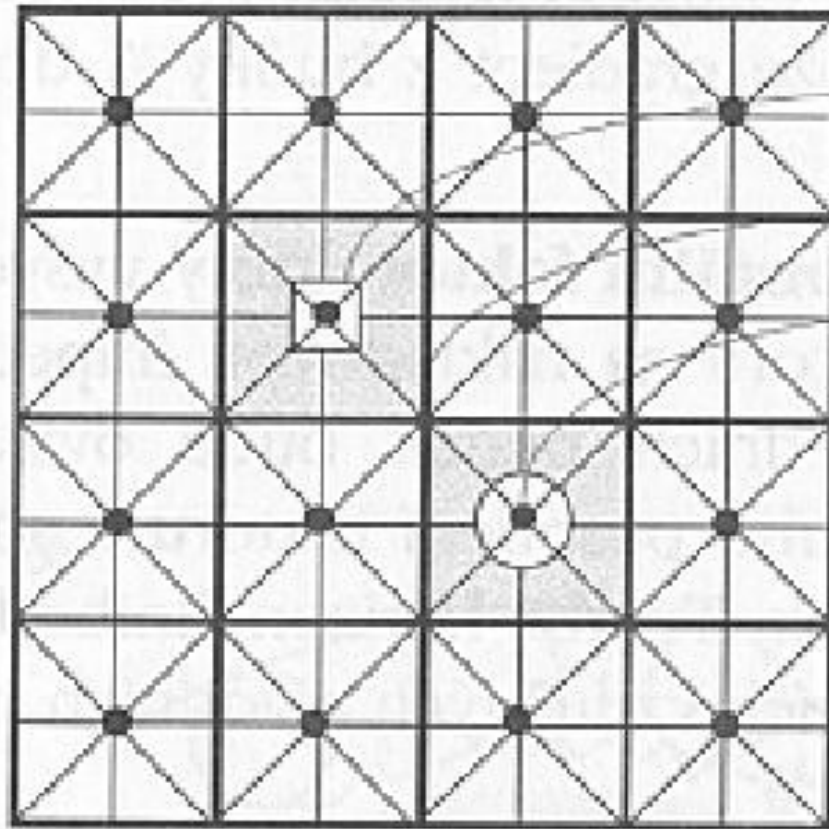
► Koncový bod (náklad 3)

$$a_2 = \frac{\text{náklad 2} + \text{náklad 3}}{2}$$

$$\text{Akumulovaný náklad} = a_1 + a_2$$

Výpočet akumulovaných
nákladů

Akumulované nákladové vzdálenosti



► Počáteční bod (náklad 1)

► a1

► Koncový bod (náklad 2)

$$a1 = \frac{1,4142 (\text{náklad 1} + \text{náklad 2})}{2}$$

Výpočet nákladů
v diagonálním
směru

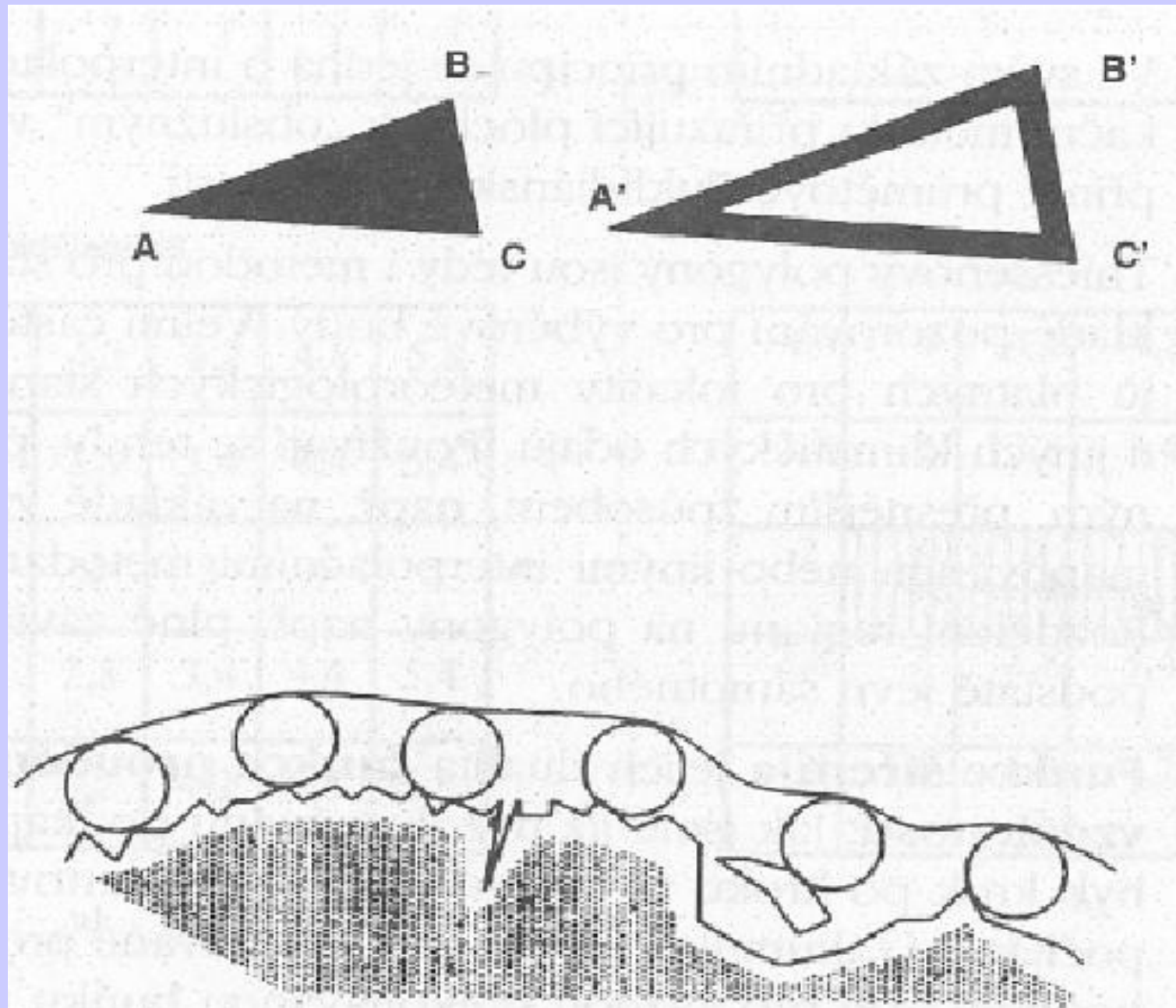
Náklady

- ◆ zohlednění směru působení odporu či nákladů
- ◆ odpory ovlivňující nákladové vzdálenosti v závislosti na směru pohybu přes danou buňku = anizotropní
- ◆ horské prostředí, proudění vody v oceánech, proudění vzdušných mas
- ◆ kromě velikosti odporu zavádíme ještě jeho směr
- ◆ vzorce pro výpočet nákladů...

Vzdálenostní zóna

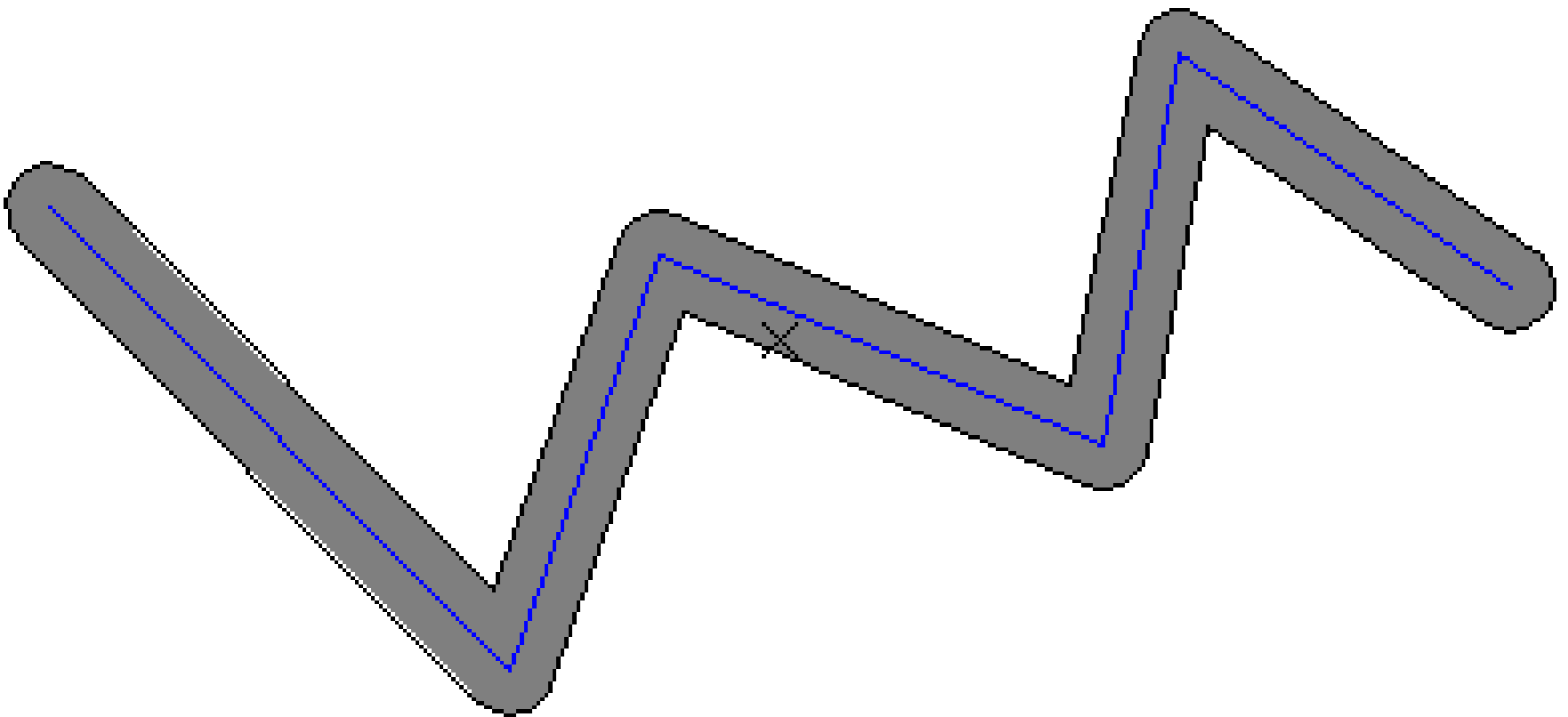
- ◆ Vzdálenostní zóna - angl. buffering. Množina bodů rastru (u vektorů podobně) majících od daného místa vzdálenost $\leq X$
- ◆ vektory - nárazníková zóna
- ◆ rastry - ve dvou krocích: krok1 - všechny buňky se ohodnotí vzdáleností, krok2 - selekce
- ◆
- ◆ GRASS: r.buffer input= output= distances=X

Nárazníková zóna



Buffery ve vektorech

- ◆ v.buffer input=linie output=bufl buffer=1000



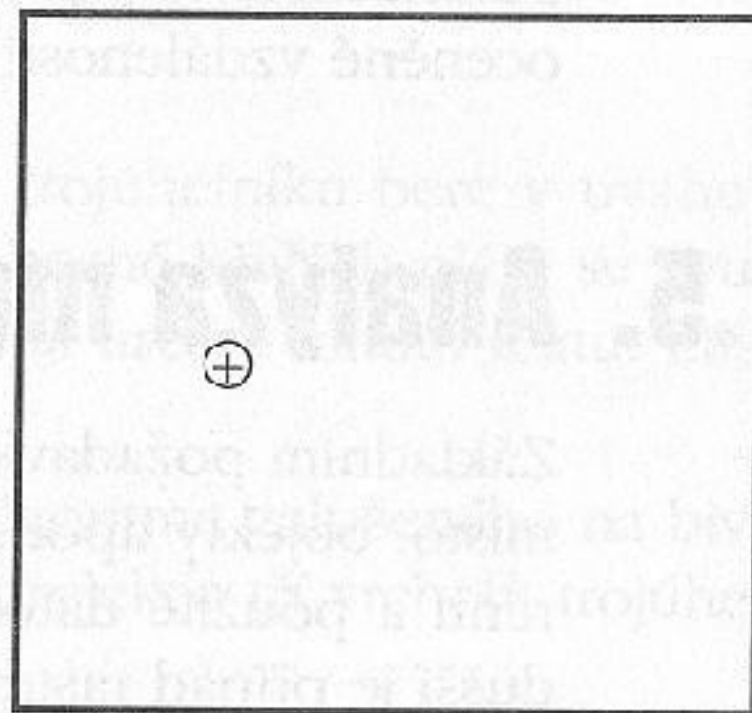
Funkce šíření, funkce proudění

- ◆ Analýzy oceněných vzdáleností.

Frikční povrch

2	2	2	1	1	1
2	2	2	1	1	1
2	2	2	1	1	1
2	2	2	1	1	1
2	2	2	1	1	1

Startovací bod



a: Funkce šíření, b: funkce proudění

- ◆ šíření – kumulativní vzdálenost z daného místa do všech směrů (nákladový prostor)
- ◆ proudění – z daného místa hledá „nejrychlejší cestu“ do bodu s potenciálem 0

Oceněné vzdálenosti

4,8	4	4,2	4,2	4,8	5,8
2,8	2	2,8	3,4	4,4	5,4
2	①	2	3	4	5
2,8	2	2,8	3,4	4,4	5,4
4,8	4	4,2	4,2	4,8	5,8

a)

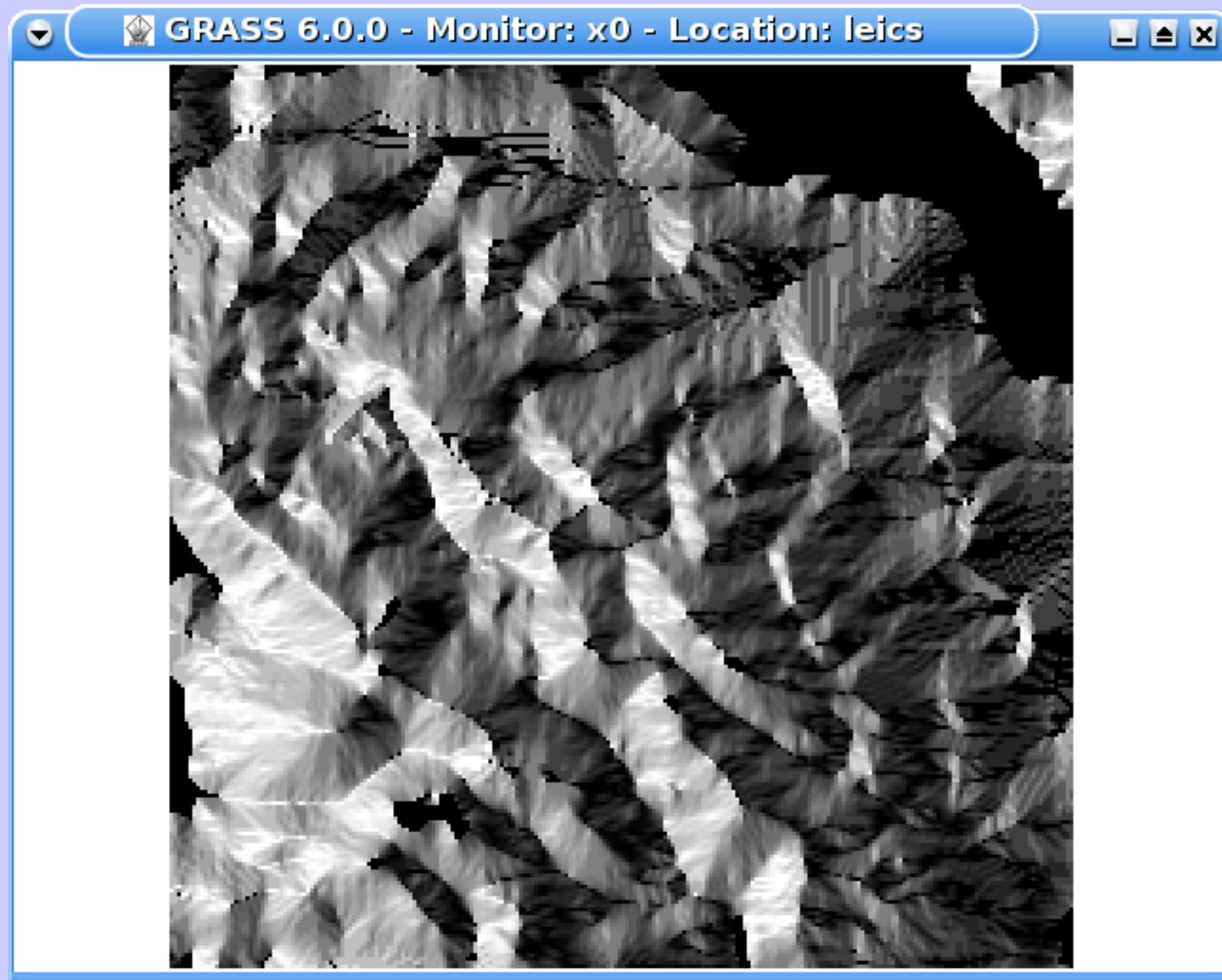
4,8	4	4,2	4,2	4,8	5,8
2,8	2	2,8	3,4	4,4	5,4
2	0	2	3	4	5
2,8	2	2,8	3,4	4,4	5,4
4,8	4	4,2	4,2	4,8	5,8

b)

Case study: kanalizační trubka

- ◆ Location: Leicester, UK
- ◆ <http://www.geog.le.ac.uk/assist/grass/seeds/gsc.html>
- ◆ Zadáním je navrhnout potrubí mezi zvýrazněnými body, minimalizace nákladů
- ◆ minimální převýšení na trase (drahé pumpy) – vést raději „down-hill“
- ◆ `r.slope.aspect elevation=topo aspect=asp1`
- ◆ `asp1` obsahuje orientace svahů

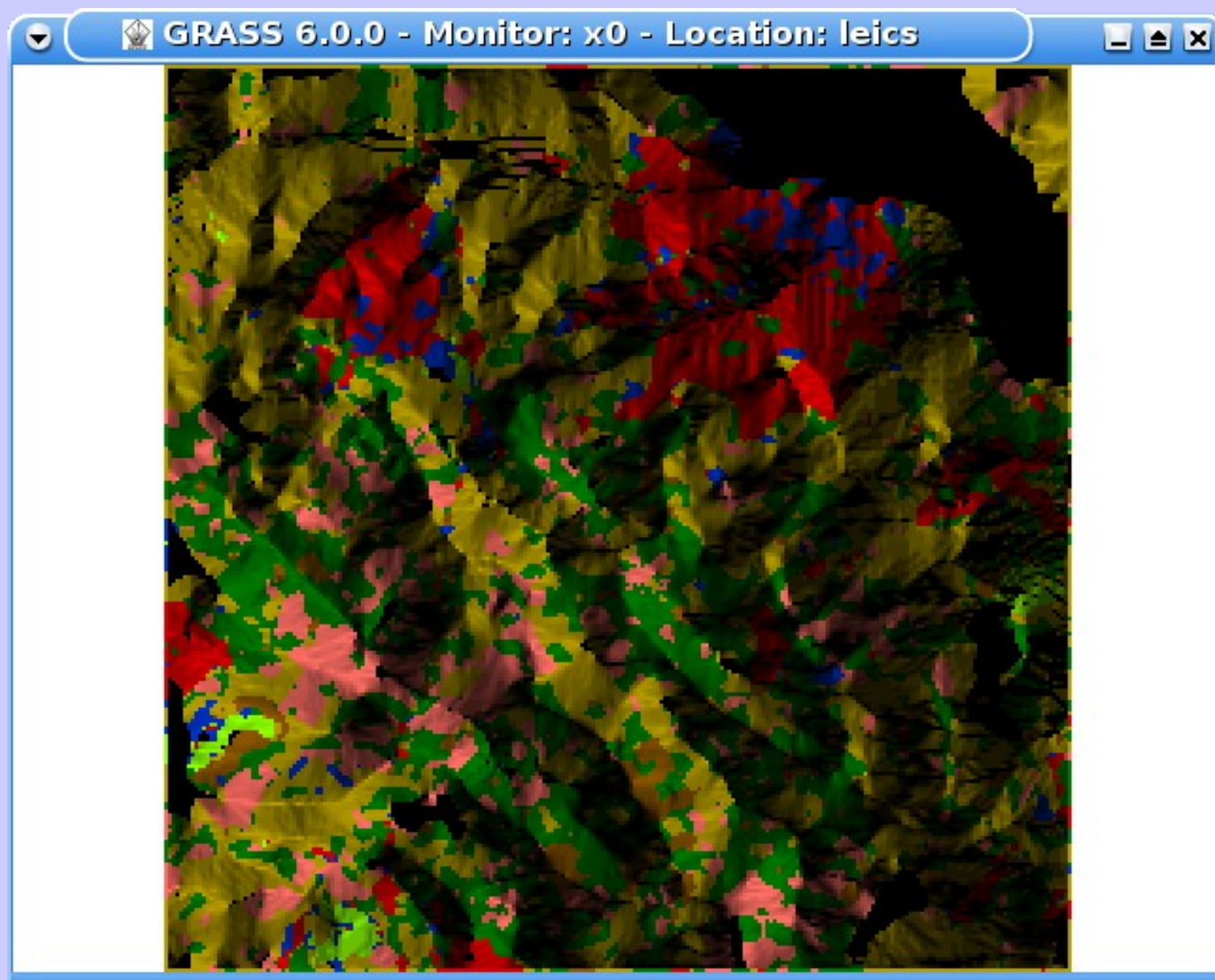
Leics: asp1



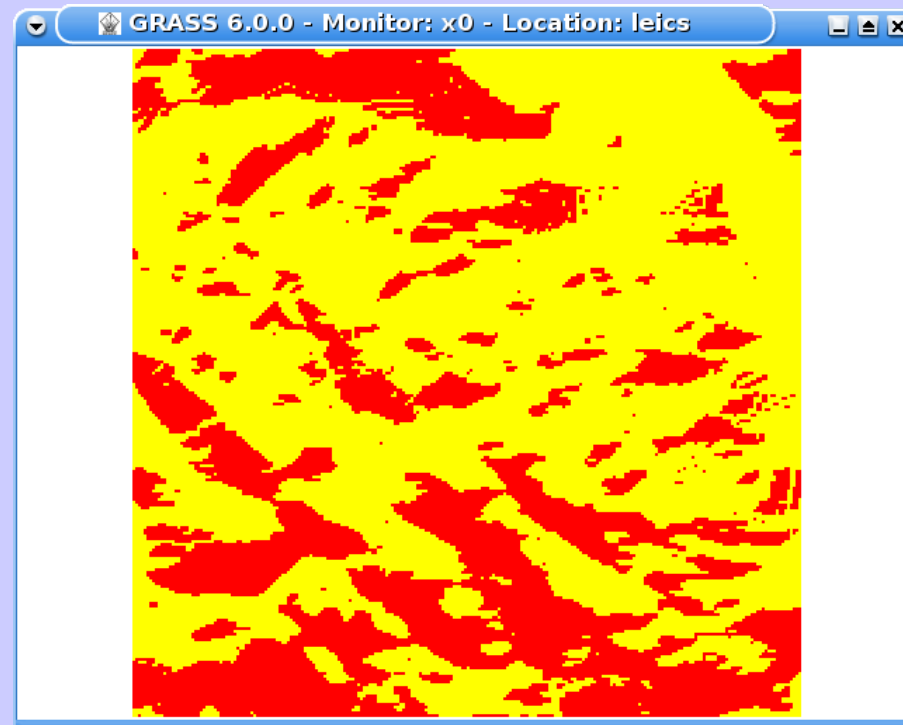
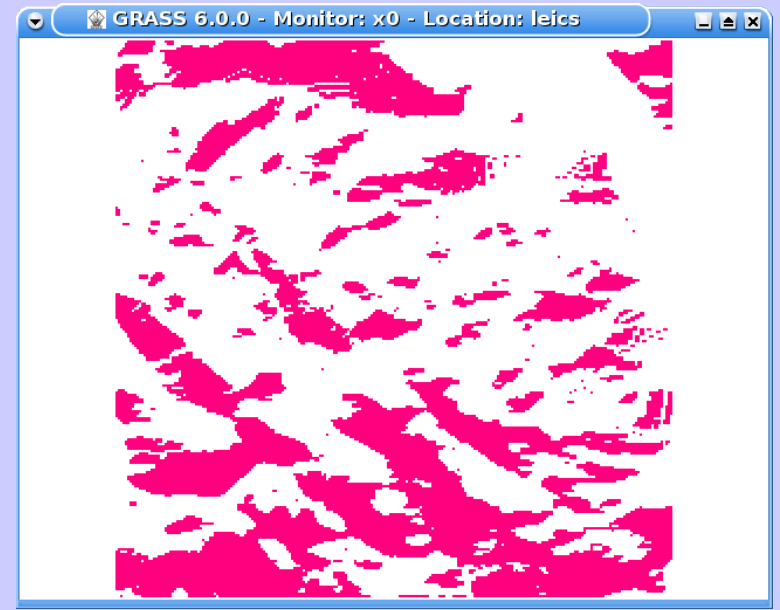
Case study: kanalizační trubka

- ◆ (omezení pracovního výřezu mapy)
- ◆ `d.rast landcov;d.his h_map=landcov i_map=asp1`
(prolnutí dvou rastrů)
- ◆ `r.rescale input=asp1 from=0,180`
`output=asp_north to=1,1`
- ◆ `r.rescale input=asp1 from=181,360`
`output=asp_south to=10,10`
- ◆ sloučení map (`r.patch`)

Zobrazení landcov přes asp1



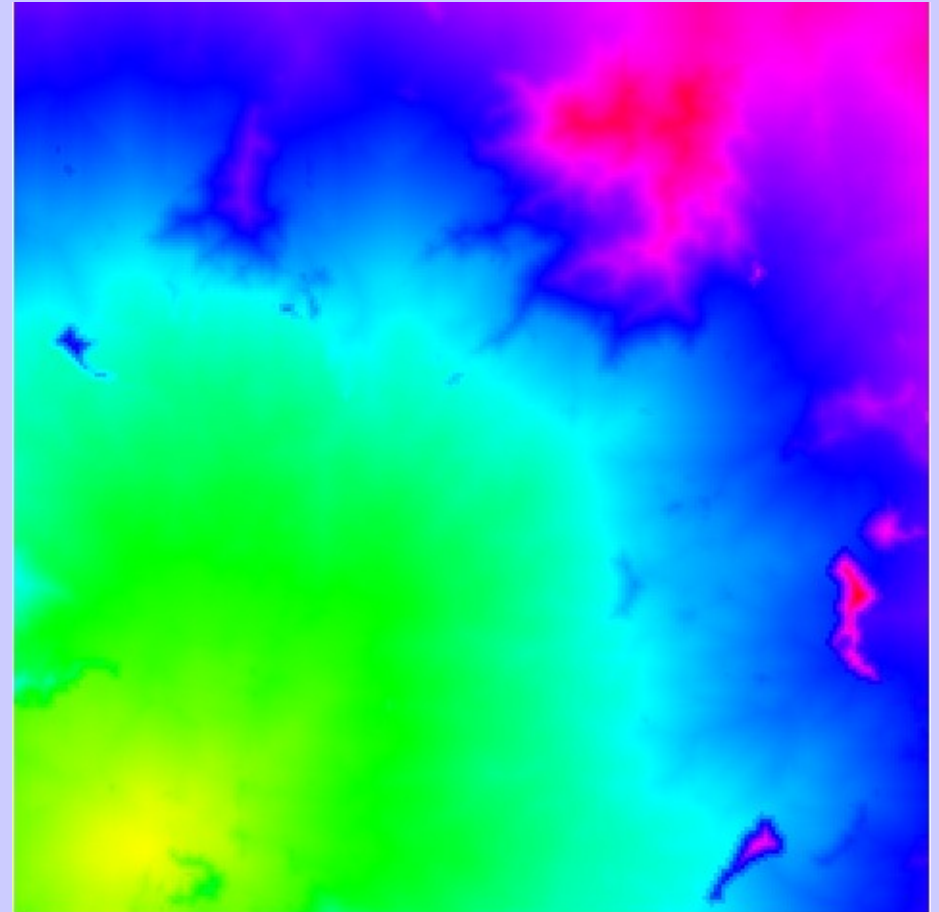
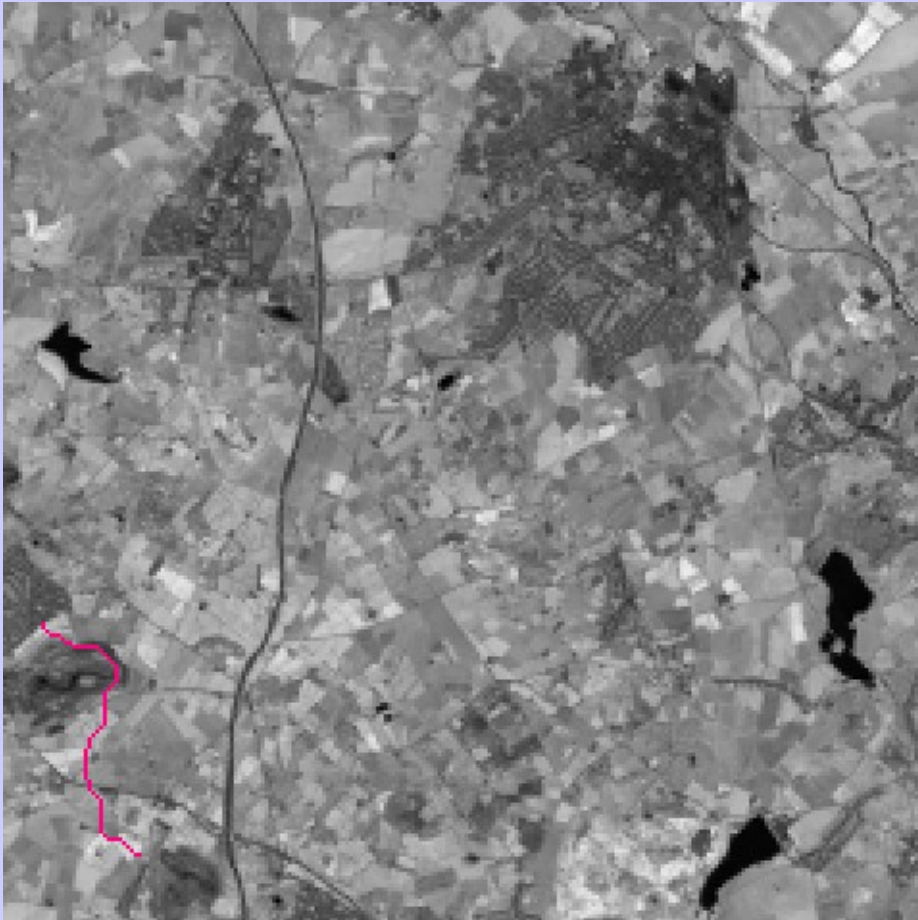
asp_north, asp_south, asp_cost



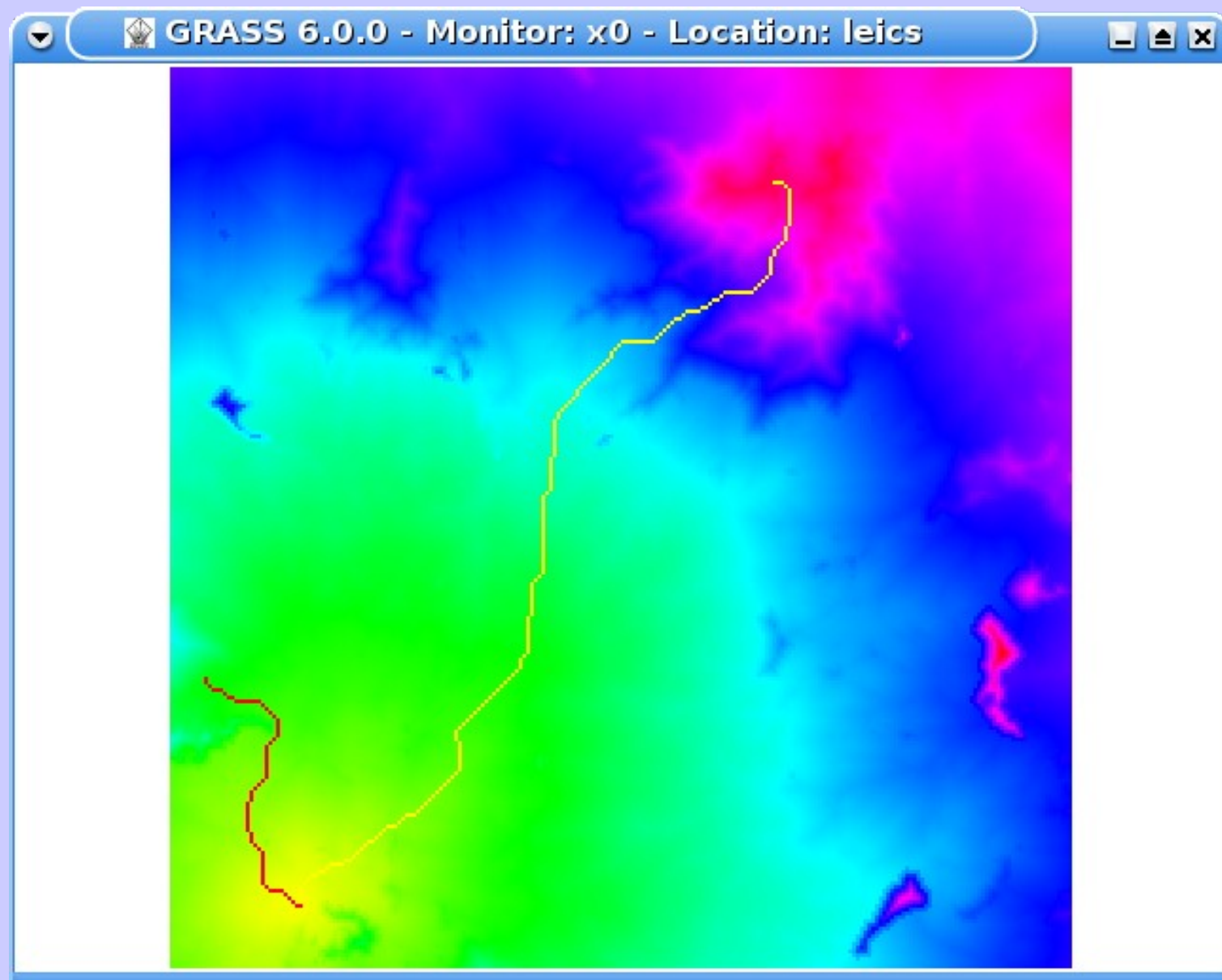
- ◆ `r.reclass input=landcov output=lan_cost` (zadaná tabulka)
- ◆ `r.mapcalc: cost = lan_cost + asp_cost`
- ◆ `r.colors map=cost color=grey`
- ◆ výpočet mapy "kumulativních nákladů" z daného bodu
- ◆ `r.cost input=cost coord=445775,310875`
`output=cost_dist`
- ◆ výpočet ideální cesty z A do B
- ◆ `r.drain input=cost_dist coord=444525,313875`
`output=pipeline`
- ◆ (r.drain taky počítá cestu řeky výškovým modelem)

lan_cost, cost_dist, pipeline

```
GRASS 6.0.0 (leics):~ > r.reclass input=landcov output=lan_cost
Enter rule(s), "end" when done, "help" if you need it
Data range is 1 to 8
> 1=1000
> 2=2000
> 3=5000
> 4=300
> 5=500
> 6=500
> 7=200
> 8=10000
> end
```



d.rast image ; d.rast -o pipeline

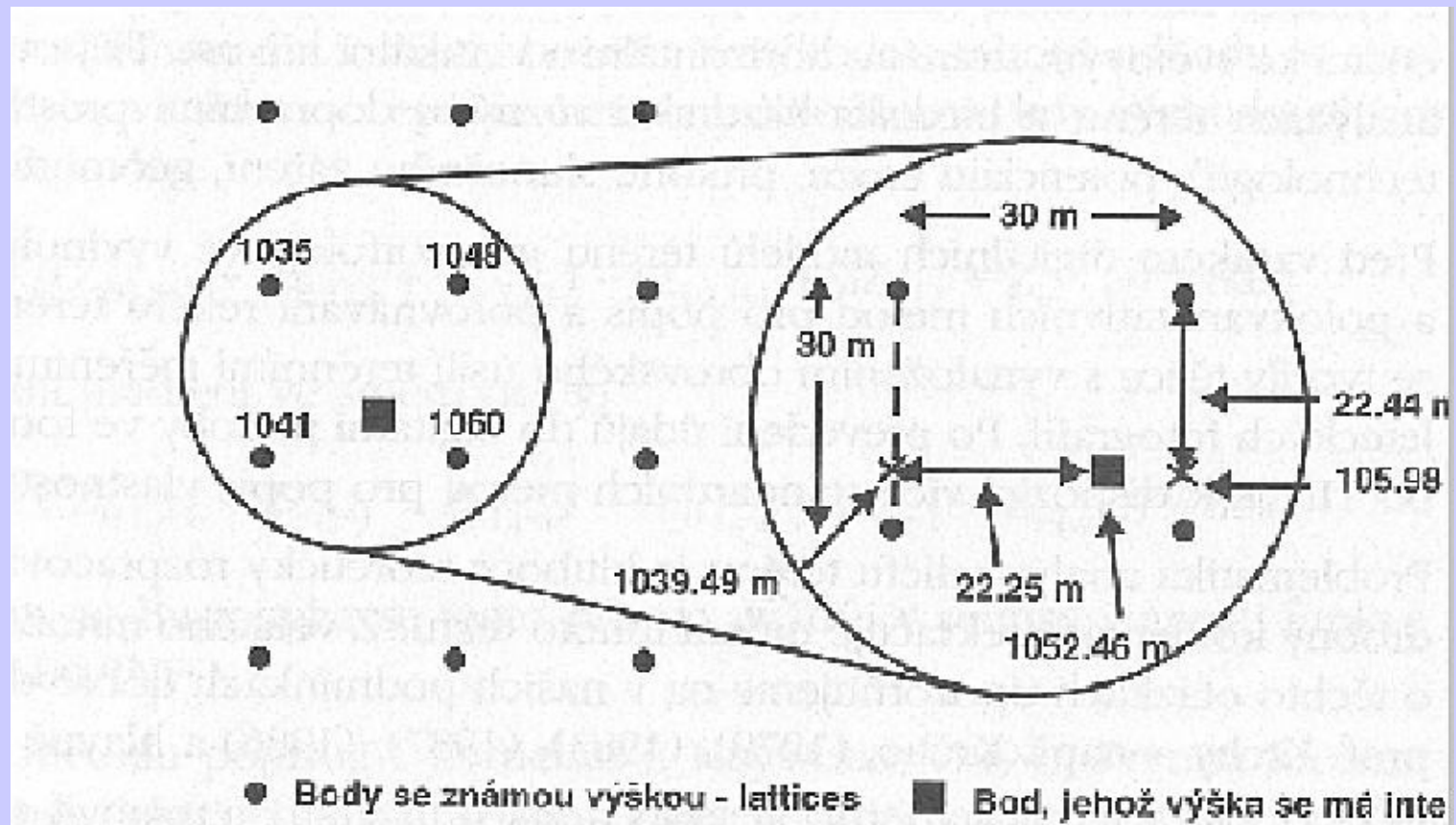


Analýza modelů terénu

- ◆ z větší části se mluví o odvozování výšky terénu
- ◆ zjednodušení: pro celou plochu buňky se uvažuje stejná výška (kategorické vyjádření výšky)
- ◆ lattices: výškový model ukládán jako množina pravidelně rozmístěných bodů s udáním výšky
- ◆ lattices = body se známou výškou
- ◆ pro lattices se používá bilineární interpolace

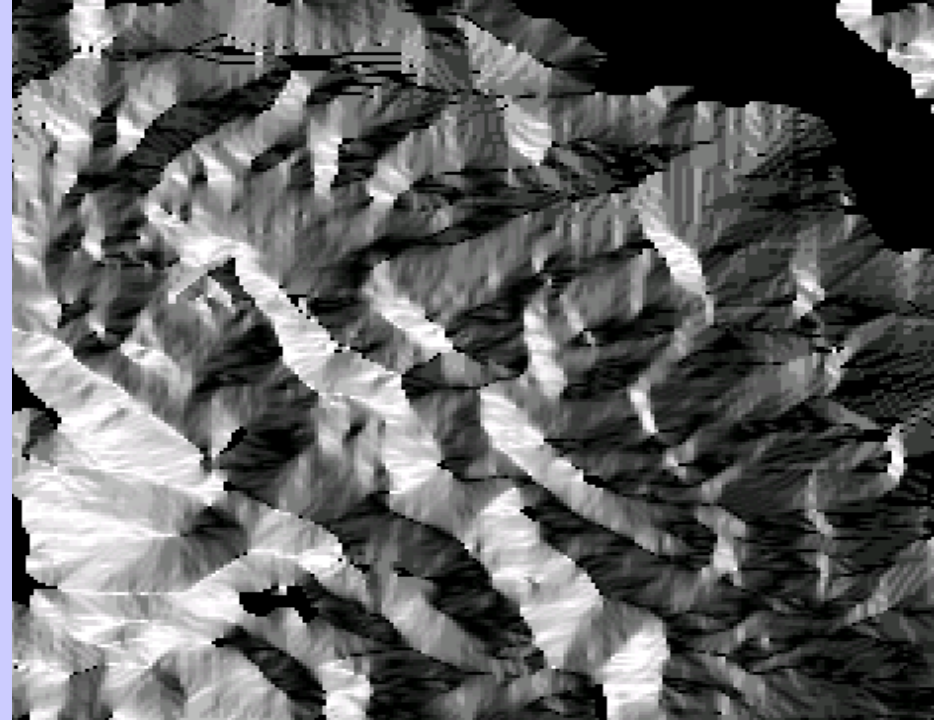
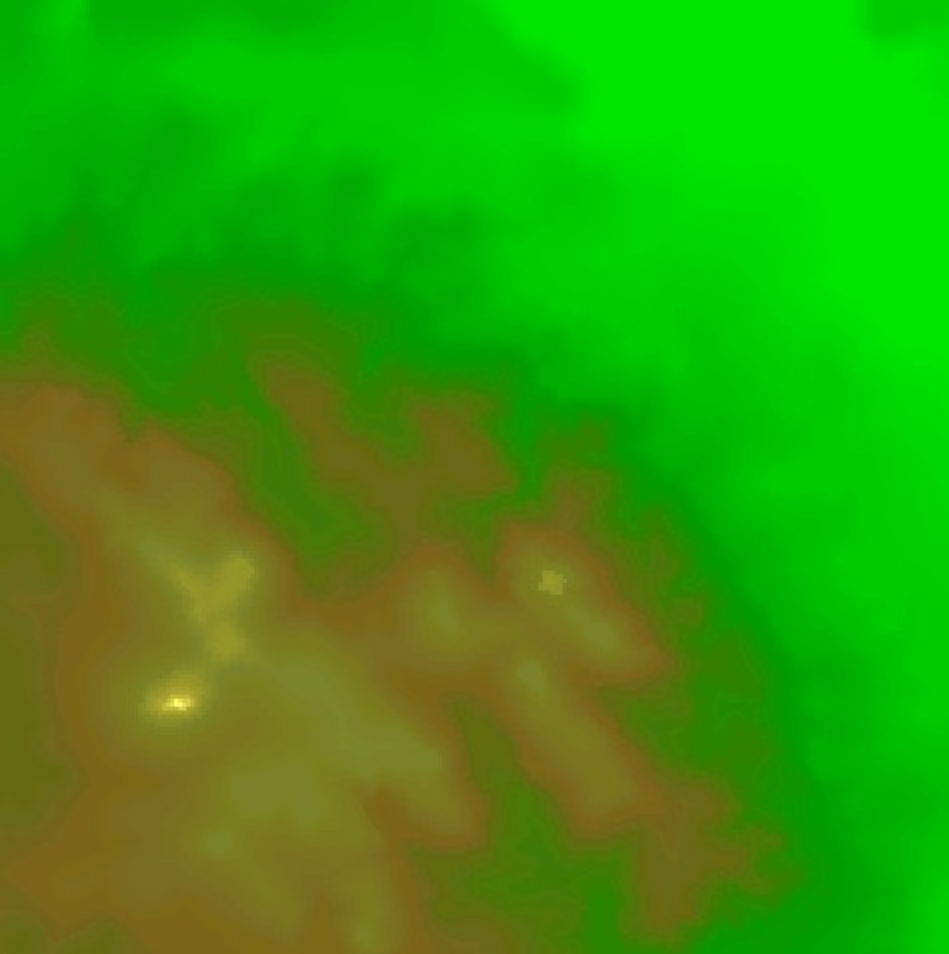
Bilineární interpolace

- ◆ $x1 = (22.4/30) * (1041 - 1035)$
- ◆ $x2 = (22.4/30) * (1060 - 1048)$
- ◆ $vys = (22.25/30) * (x2 - x1) + x1$
- ◆ $r.surf.*$



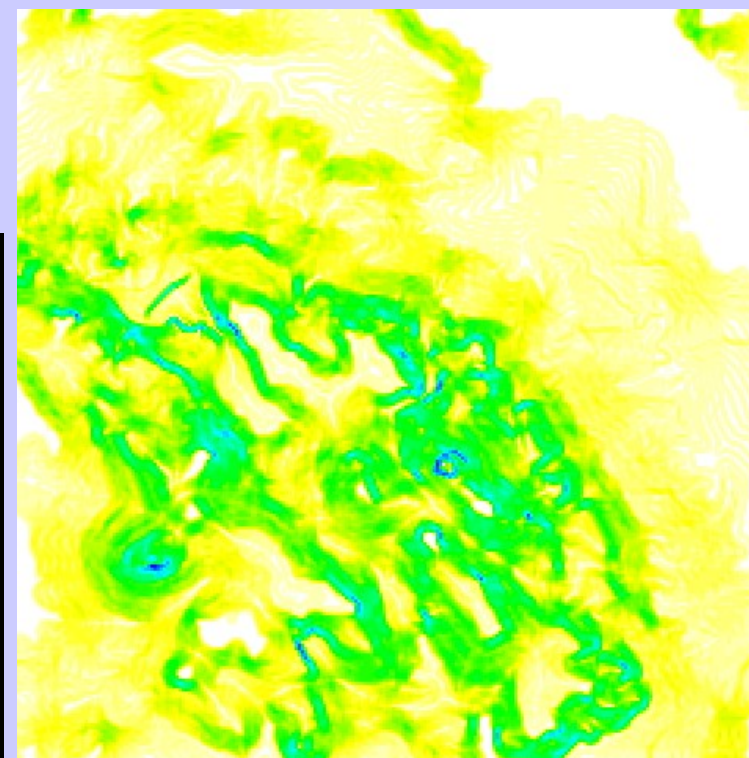
Analýza modelů terénu - sklon

- ◆ Anglicky "slope".
- ◆
- ◆ velmi užitečná informace pro analýzu krajiny
- ◆ definice: sklon je tangenta roviny povrchu
- ◆ má dvě složky:
 - ◆ gradient G - maximální poměr změny výšky
 - ◆ aspekt A - azimutová orientace této max. změny výšky
- ◆ GRASS: `r.slope.aspect`



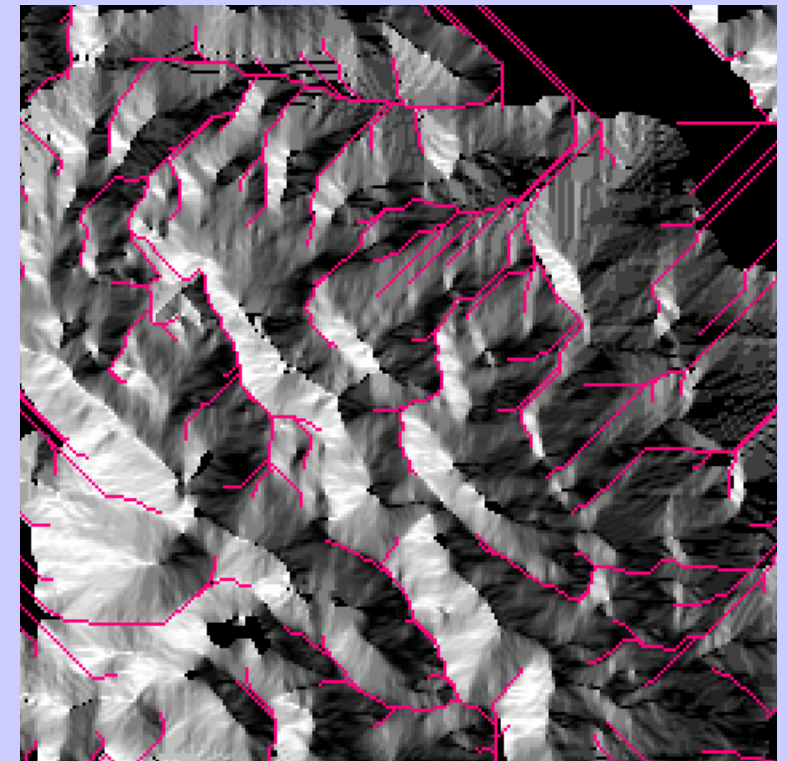
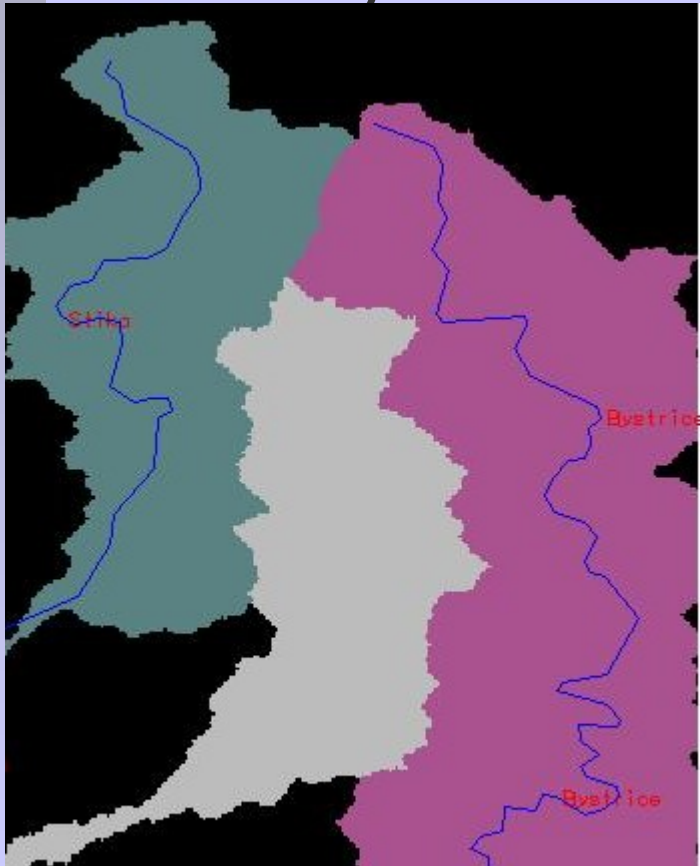
Type of Map: raster Number of Categories: 255
Data Type: FCELL
Rows: 240
Columns: 240
Total Cells: 57600
 Projection: UTM (zone 0)
 N: 322000 S: 310000 Res: 50
 E: 456000 W: 444000 Res: 50
Range of data: min = 0.000000 max = 14.781473

Data Source:
 raster elevation file topo



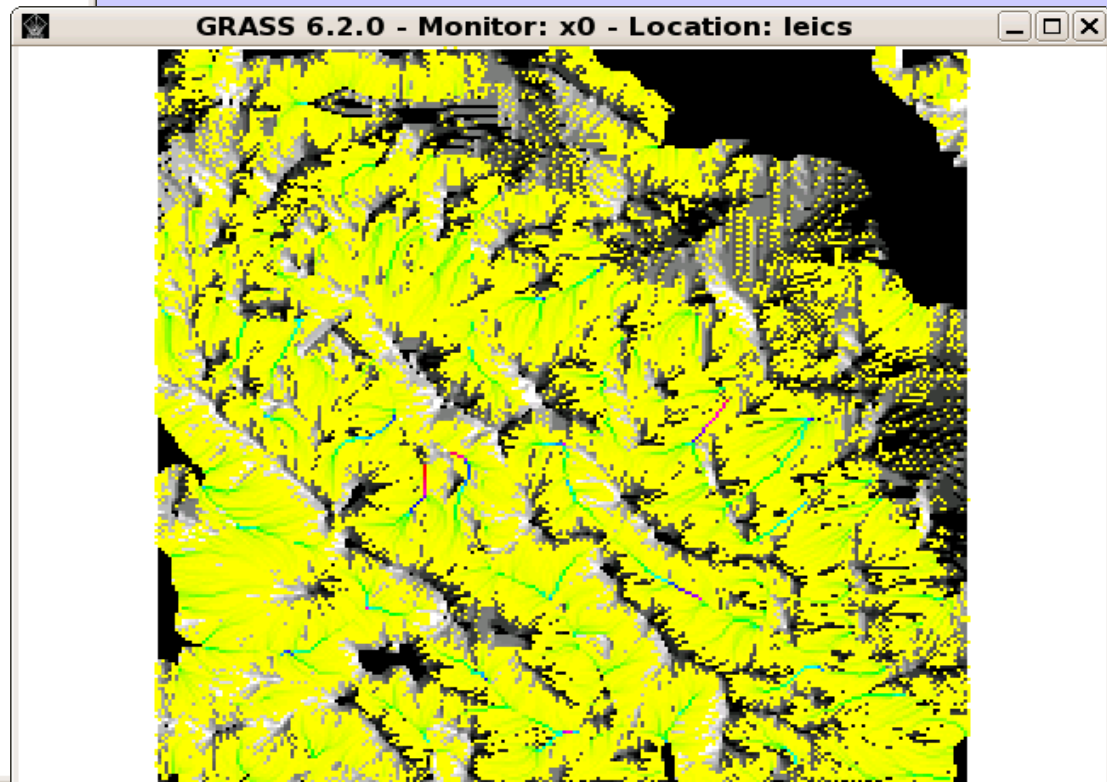
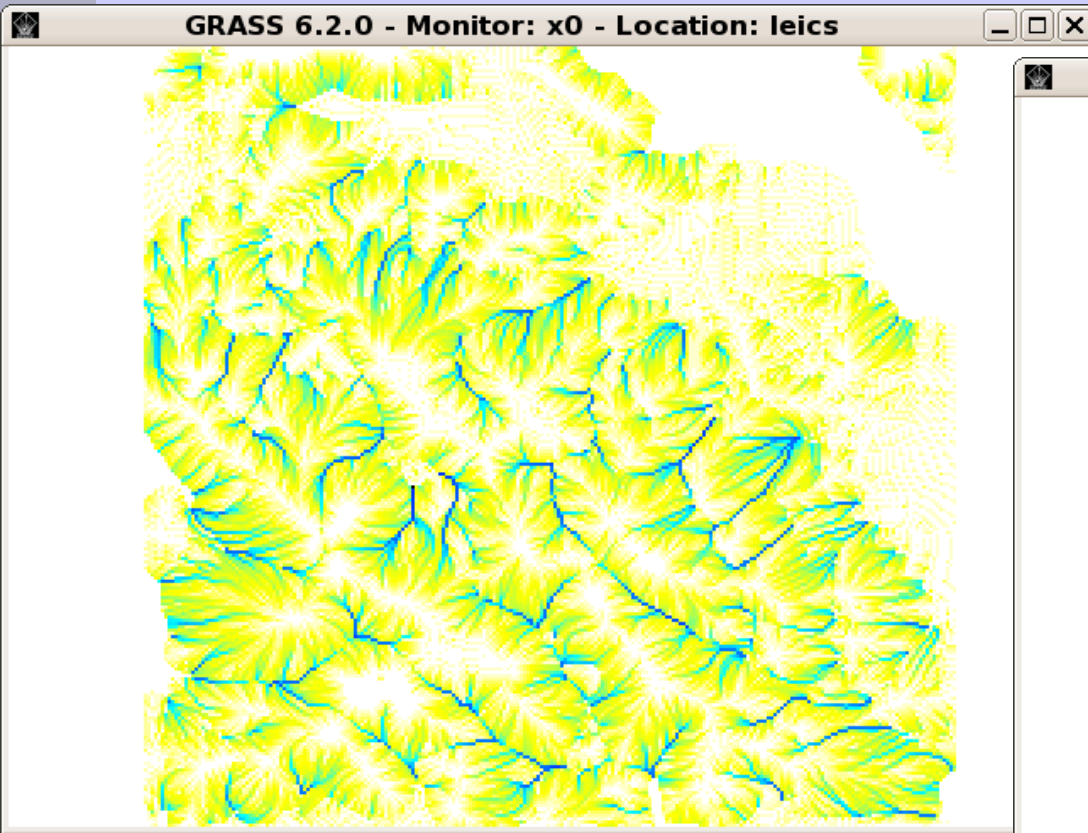
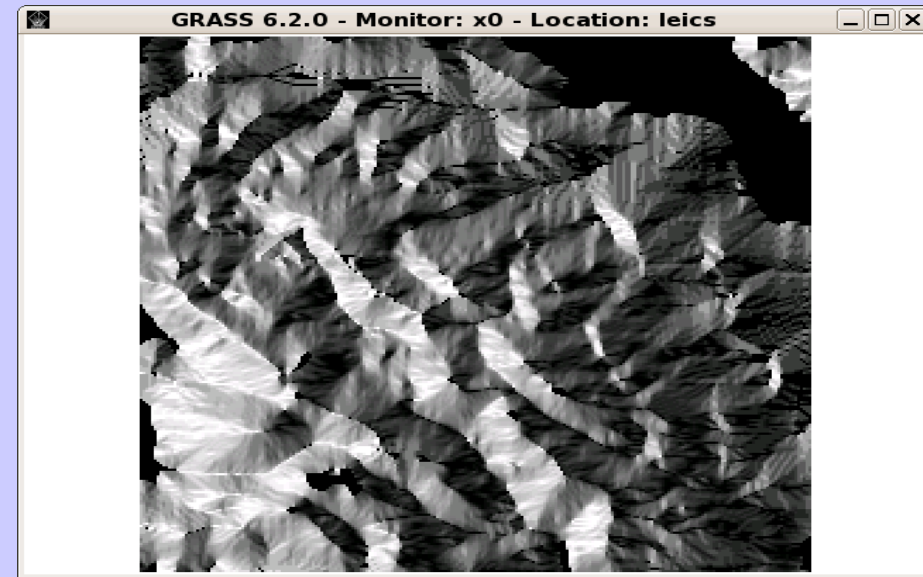
Sklon

- ◆ identifikace hřebenů, údolnic, povodí, bezodtoké oblasti
- ◆ modelování eroze - výpočet délek svahů a geomorfologických forem terénu
- ◆ Analýza viditelnosti – obrs.
- ◆ Analýza osvětlení - r.sun

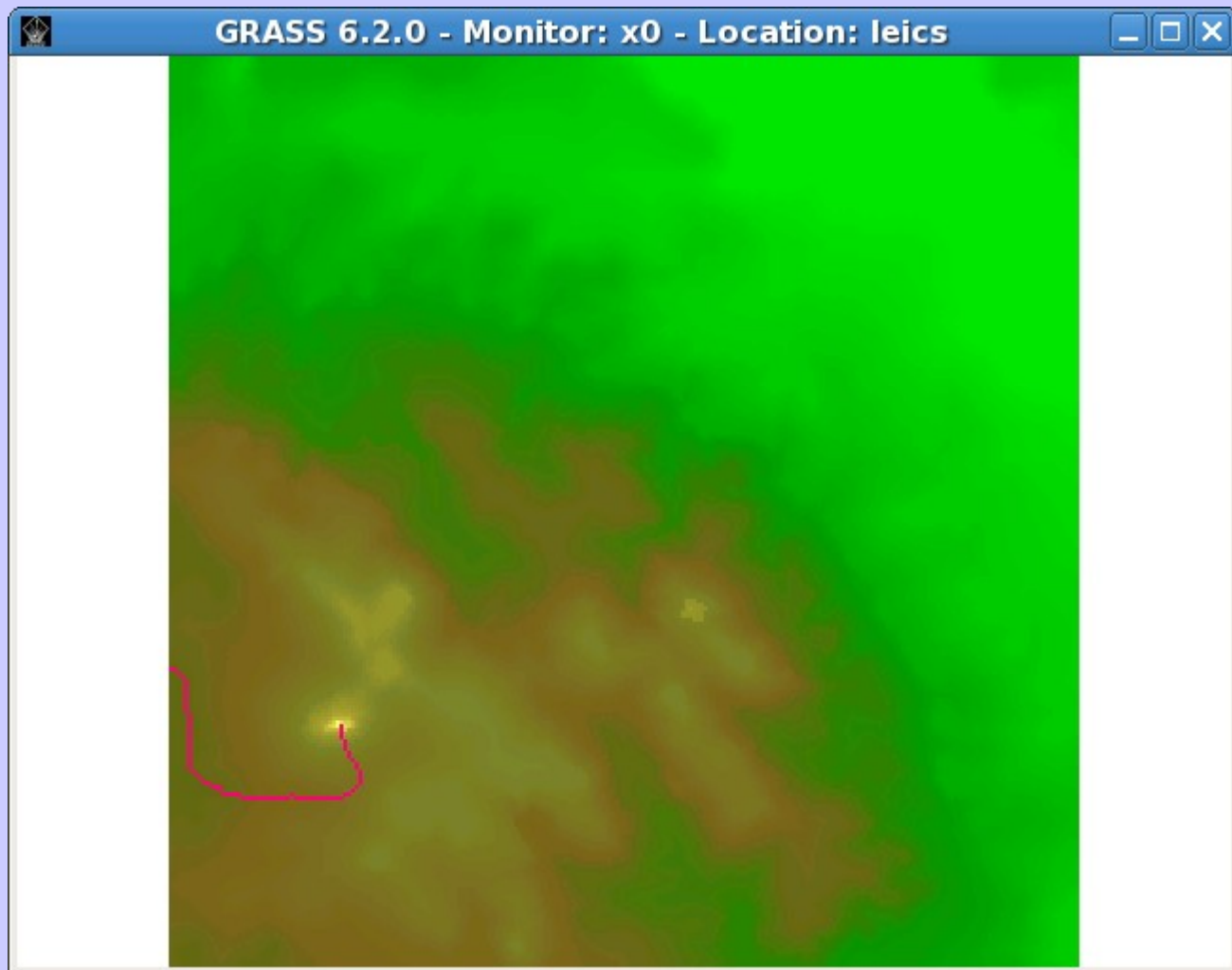


r.flow

- ◆ Ze DEM zjistí linie stoku ze svahů
- ◆ r.drain



r.drain



Auto: ☒ Clear ☒ Draw

Feature:

View: ☒ eye ☐ center ☐ fly none

N

W

E

S

Height

640.00

zexag

30.77

LOOK

here

center

top

cancel

RESET

perspective

40.0

twist

0.0

Lighting Panel

☐ Follow ViewPoint

☒ Show Model

Brightness

0.8

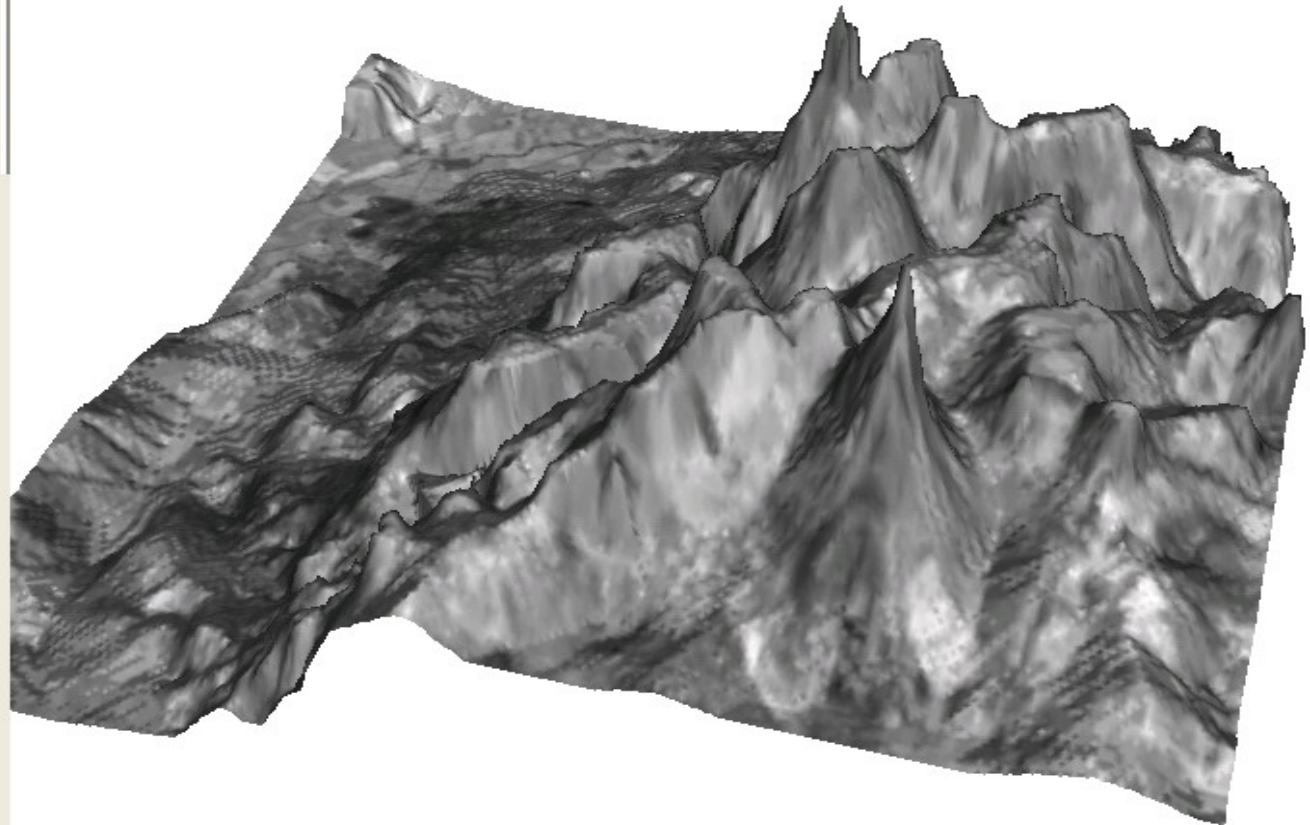
Ambient

0.2

Height

0.8

W



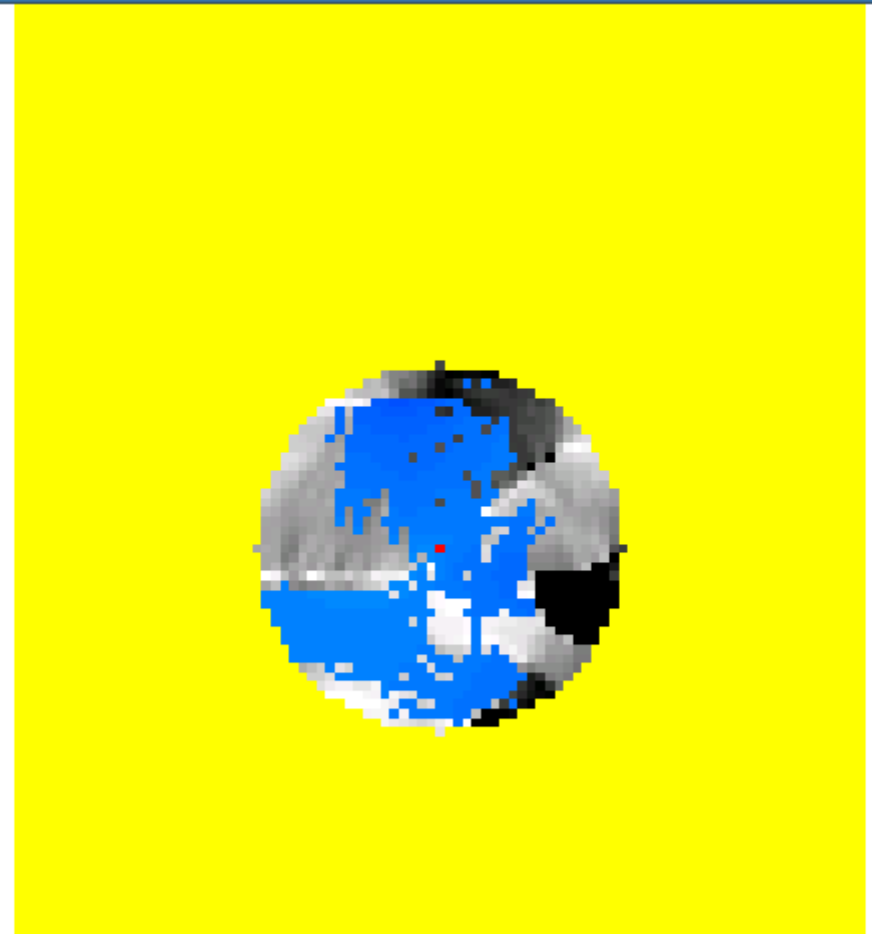
*r.lo*s

◆ Souřadnice, okolí, DEM

GRASS 6.2.0 - Monitor: x0 - Location: leics



GRASS 6.2.0 - Monitor: x0 - Location: leics

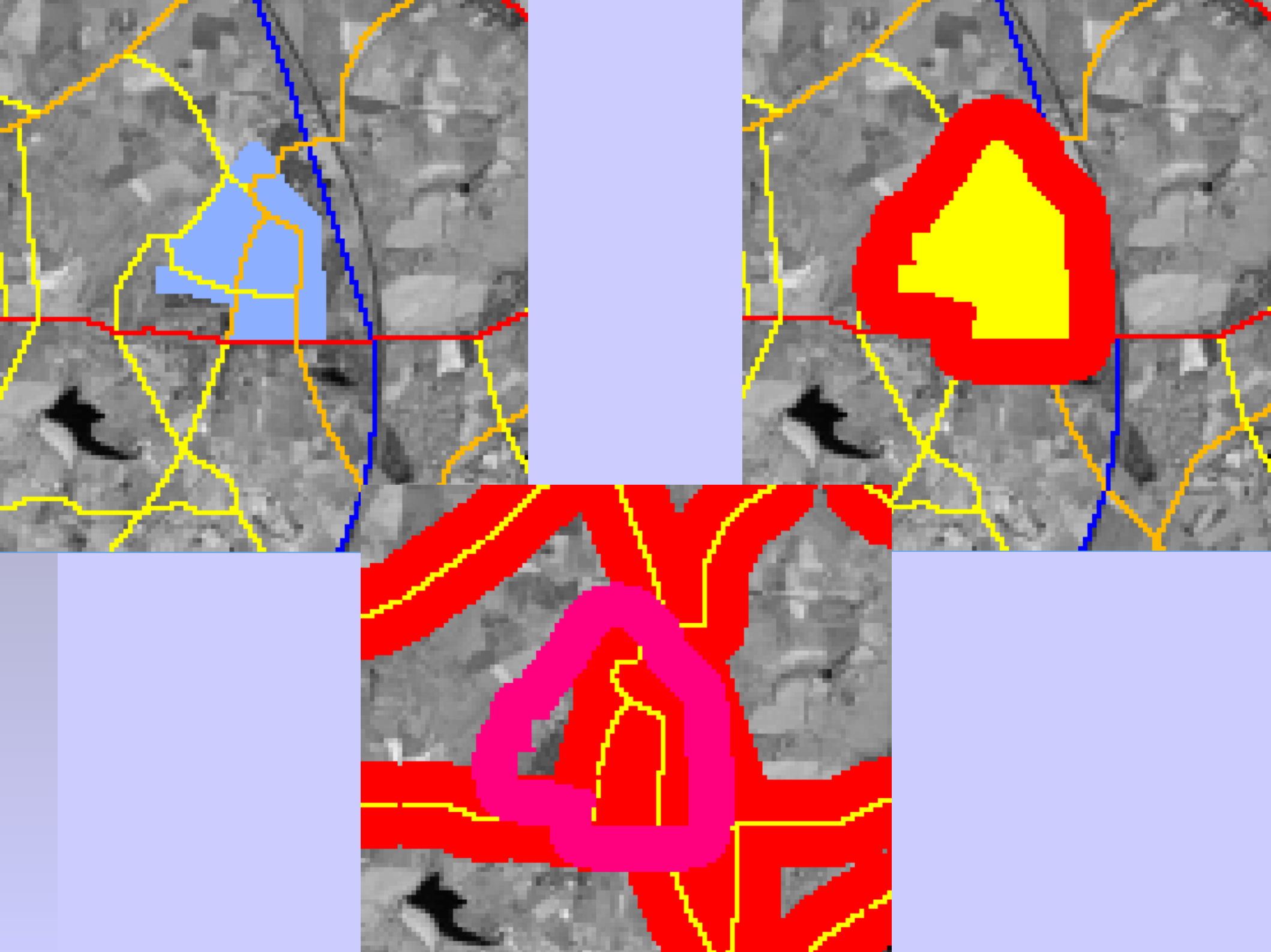


Case study: sportovní hřiště

- ◆ 500m od města Shepshed (nedaleko pro lidi)
- ◆ 450m od dálnice a od silnic tříd A a B (dostupnost autem)
- ◆ sklon pozemku menší než 2 st.
- ◆ na pozemku typu zemědělský, typ III (cena)
- ◆ plocha aspoň 2.5 ha
- ◆ mimo přímou viditelnost z obydlených oblastí

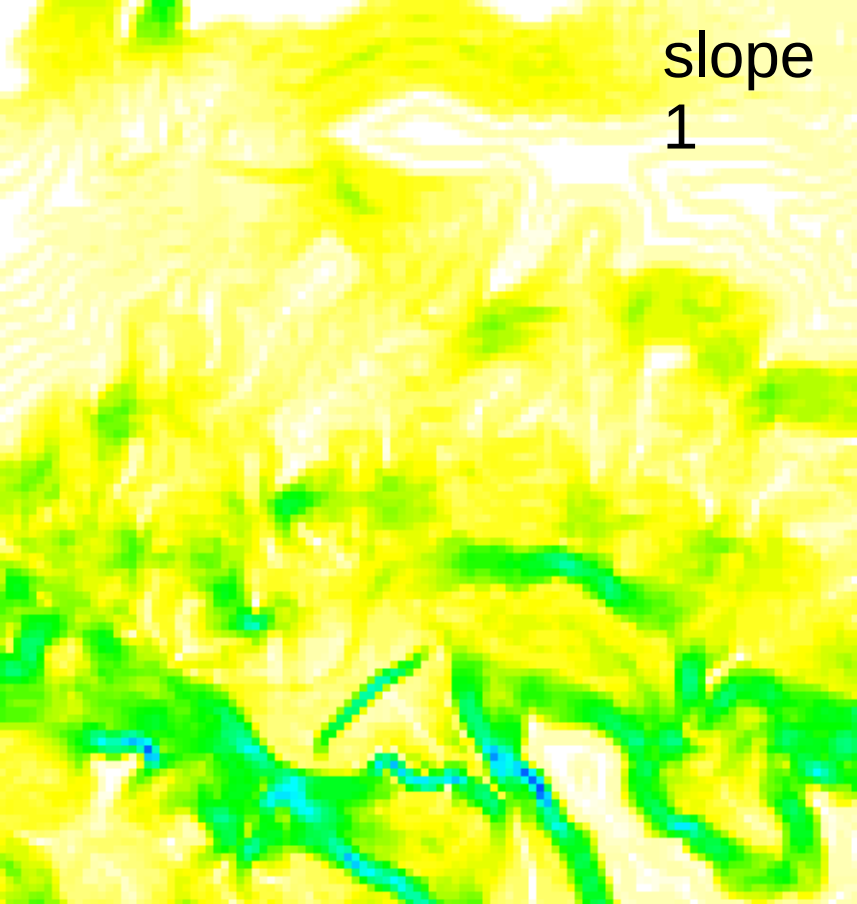
Case study: sportovní hřiště

- ◆ g.region se nastaví na oblast kolem města (GRASS počítá jenom viditelný region)
- ◆ r.buffer urban distances=500
output=urbanbuf
- ◆ echo "2=1" | r.reclass input=urbanbuf
output=urbanbuf2
- ◆ echo "3 9 11=1" | r.reclass input=roads
output=mainroads
- ◆ r.buffer mainroads distance=450
output=mainroadbuf



Case study: sportovní hřiště

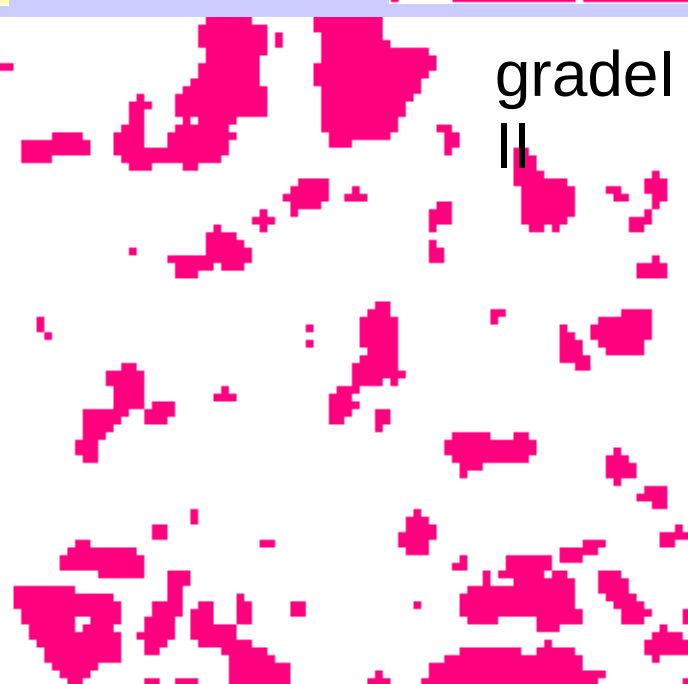
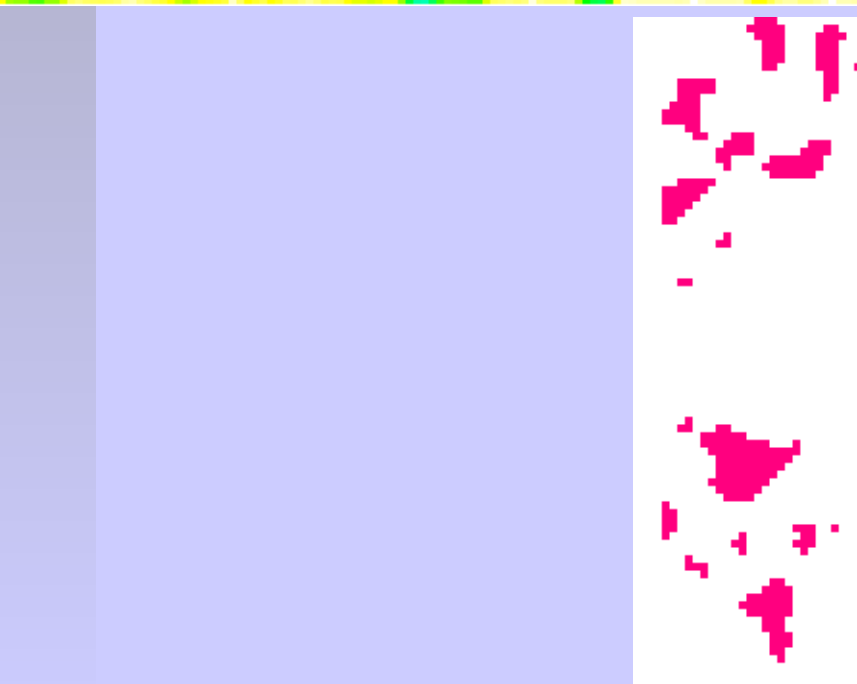
- ◆ `echo "2=1" | r.reclass ...`
- ◆ `r.slope.aspect elevation=topo slope=slope1`
- ◆ `r.reclass (0 1 2 = 1) - flat`
- ◆ identifikace půdy - reklasifikace z landcov (kategorie 6,7) - `gradelll`
- ◆ konstrukce mapy (`r.mapcalc`)
- ◆ `mapcalc> sites = urbanbuf2 * mainroadbuf2 * flat * gradelll`



slope
1

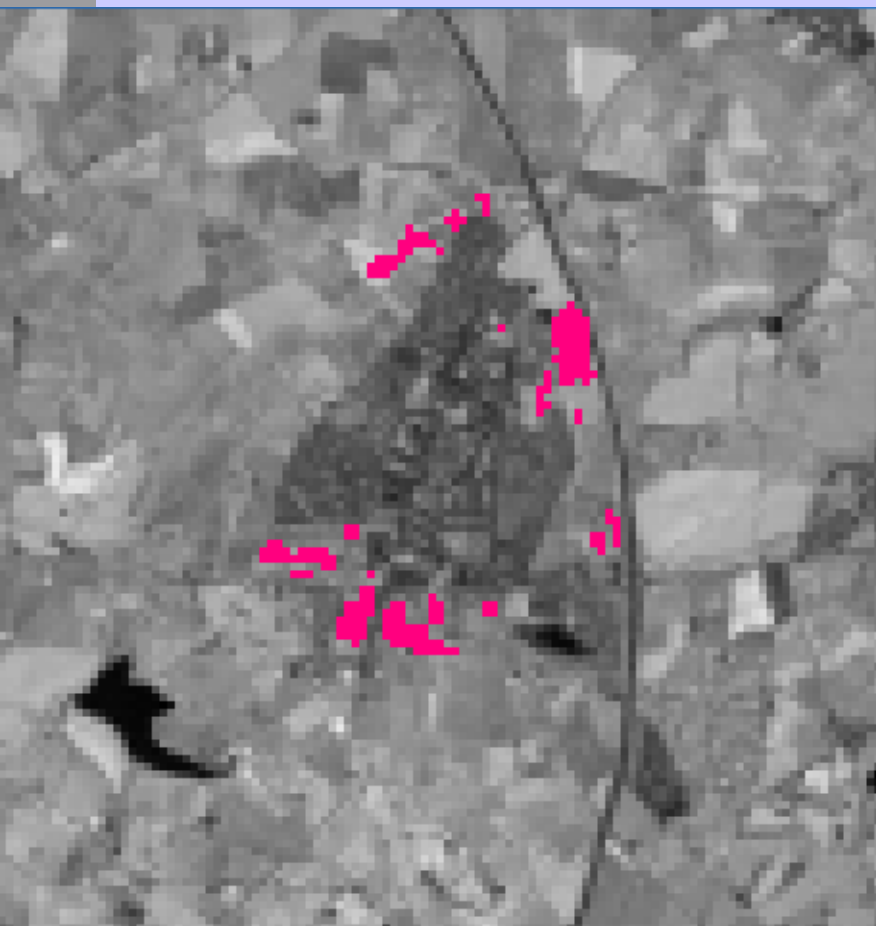


slope<=
2



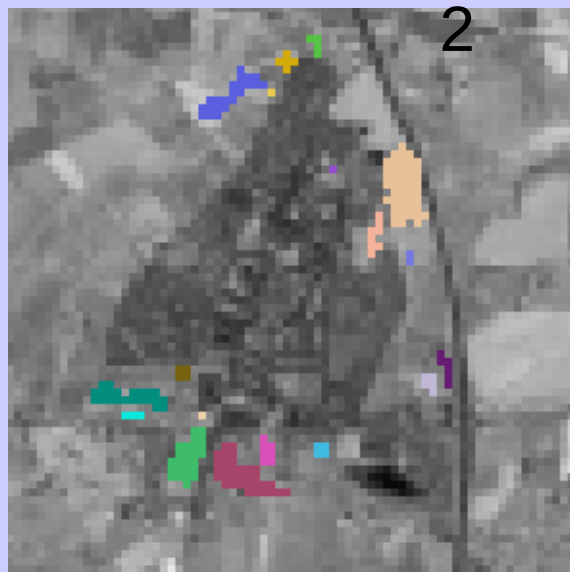
gradel
II

sites



sites

2



sites3



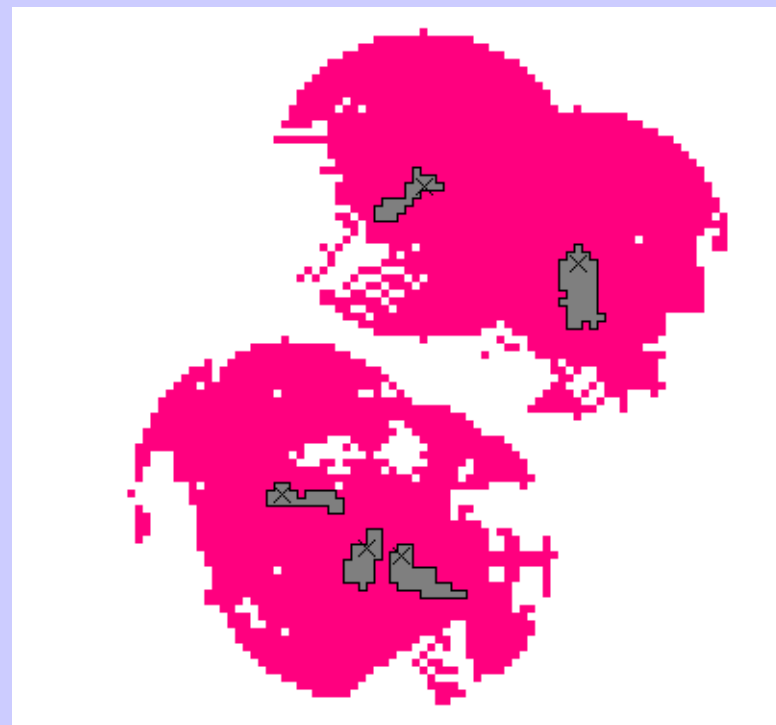
Category Information		hectares
# description		
1	.	1.00000
2	.	1.25000
3	.	5.50000
4	.	0.25000
5	.	11.25000
6	.	0.25000
7	.	2.00000
8	.	0.50000
9	.	1.50000
10	.	1.00000
11	.	1.25000
12	.	5.50000
13	.	0.75000
14	.	0.25000
15	.	6.25000
16	.	1.75000
17	.	8.25000
18	.	1.00000
* no data.	.	3400.50000

Case study: sportovní hřiště - plochy

- ◆ sites...plochy splňující první 4 podmínky
- ◆ r.clump sites output=sites2 (rozdělí vrstvu na souvislé plochy a dá jim odlišné kategorie)
- ◆ r.report sites2 units=hect (vypíše obsahy plošek)
- ◆ vyberou se ty s plochou větší než 2.5 ha, reklasifikace - sites3
- ◆ převod sites3 na vektory (polygony) - r.to.vect input=sites3 output=vect_sites feature=area

Case study: sportovní hřiště - Analýza viditelnosti

- ◆ vybereme souřadnice pro umístění sportovní haly, výška 10m
- ◆ chceme znát části "popln" (do 1km), které mají přímou viditelnost na halu
- ◆ ručně zadáme polohy místa pozorovatele, získáme 5 map
- ◆ `r.los input=topo output=v... coordinate=x,y
obs_elev=10 max_dist=1000`
- ◆ výběr lokace
- ◆ `r.report popln units=cell`



r.example

```
for (row = 0; row < nrows; row++) {
    CELL c;
    FCELL f;
    DCELL d;

    if (verbose)
        G_percent(row, nrows, 2);

    /* read input map */
    if (G_get_raster_row(infd, inrast, row, data_type) < 0)
        G_fatal_error(_("Could not read from <%s>"), name);

    /* process the data */
    for (col = 0; col < ncols; col++) {
        /* use different function for each data type */
        switch (data_type) {
        case CELL_TYPE:
            c = ((CELL *) inrast)[col];
```