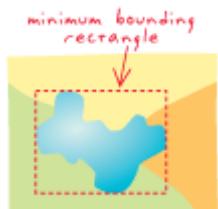


Z8154 Programování v geoinformatice (Podzim 2013)
CV03 Cykly

Úlohy (odevzdávacia otvorená do 14.10. 20:00)

1. MBR

V poli **polyg** je pomocou súradnicových párov uložený jednoduchý polygón (oblasť leží v okolí Jakubského námestí v Brne). Napíšte funkciu **mbr()** ktorá určí súradnice (ide o 4 hodnoty), ktoré definujú minimálny ohraňujúci štvoruholník k danému polygónu [1].



```
polyg=[(16.608446324830002, 49.196211250850006), (16.60875024329,  
49.19631522302001), (16.60878623328, 49.196495174800006), (16.608566293090004,  
49.19691106287), (16.60892619482, 49.197071018920006), (16.608758315190002,  
49.19831075891), (16.609014244750004, 49.19838659082), (16.60912799292,  
49.197362867070005), (16.60925121924, 49.196832046749996), (16.60915080115,  
49.19621858301), (16.60861244482, 49.19223845483), (16.60835967322,  
49.192575483030005), (16.60868248328, 49.194462315310005), (16.608446324830002,  
49.196211250850006)]
```

možné riešenie:

```
def mbr(polyg):  
    x_min = min(polyg)[0]  
    x_max = max(polyg)[0]  
    y_min = min(polyg, key=lambda x: x[1])[1]  
    y_max = max(polyg, key=lambda x: x[1])[1]  
  
    return [x_min, x_max, y_min, y_max]
```

toto riešenie využíva tzv. lambda (anonymnú) funkciu, ktorú som spomíнал len okrajovo, viac o lambda funkciách:

http://www.secnetix.de/olli/Python/lambda_functions.hawk
http://pythonconquerstheuniverse.wordpress.com/2011/08/29/lambda_tutorial/
http://www.diveintopython.net/power_of_introspection/lambda_functions.html

2. Prienik obdĺžníkov

Vytvorte funkciu **prunik()** ktorá zistí, či dva zadané MBR (minimálne ohraničujúce štvoruholníky) majú spoločný prienik alebo či sú od seba izolované. Vstupom budú dva polia so štyrmi súradnicami, ktoré definujú daný štvoruholník.

mbr1[x1, x2, y1, y2]
mbr2[x1, x2, y1, y2]

možné riešenie:

```
def prunik(mbr1, mbr2):
    if (((mbr1[0]>=mbr2[0]) and (mbr1[0]<=mbr2[1])) or ((mbr1[1]>=mbr2[0])
        and (mbr1[1]<=mbr2[1]))) and (((mbr1[2]>=mbr2[2]) and
        (mbr1[2]<=mbr2[3])) or ((mbr1[3]>=mbr2[2]) and (mbr1[3]<=mbr2[3]))):
        return True
    else:
        return False
```

Dobrovoľne: pripíšte funkciu **prunik_pl()**, ktorá vypočít plochu prieniku dvoch MBR ak existuje.

3. Veková pyramída

Nasledujúci zoznam predstavuje zjednodušenú vekovú štruktúru ČR. Percentuálne hodnoty sú zoradené od najstaršej vekovej skupiny (65 a viac) po najmladšiu (0-14). Prvá hodnota z dvojice predstavuje mužov, druhá hodnota ženy.

vek struktura=[(13,19),(7,7),(7,7),(6,6),(7,6),(7,6),(9,8),(9,8),(7,7),(7,6),(6,5),(15,14)]

Vytvorte funkciu **vek_pyramida()**, ktorá na základe dát zo zoznamu **vek_struktura** vykreslí jednoduchú vekovú pyramídu. Ako základ môžete použiť program pyramída z cvičenia. Výstup môže vyzerat' na príklad takto:

možné riešenie:

```
def vek_pyramida(data):
    odsad = max(data)[0]
    odsad = odsad +3
    for i in data:
        print (odsad - i[0])*" "+i[0]*"|"+" "+i[1]*"|"
```

odsadenie môžeme vykonať aj pomocou funkcie **rjust()**

Odosadzajte v jednom textovom súbore do odevzdávarej (deadline 14.10. 20:00).

Dobrovoľné úlohy:

Súbory úloh 7, 9, 12, 14 na stránke <http://www.codecademy.com/tracks/python>

Zdroje:

[1]<http://support.esri.com/en/knowledgebase/GISDictionary/term/minimum%20bounding%20rectangle>