

1. (5 bodov) Vytvorte funkciu `inverzná(R)`, ktorá vytvorí a vráti reláciu R^{-1} , danú predpisom

$$(x, y) \in R^{-1} \Leftrightarrow (y, x) \in R.$$

```
>>> B=[(1,2),(3,4),(0,1)]
>>> inverzna(B)
[(2, 1), (4, 3), (1, 0)]
```

2. (5 bodov) Vytvorte funkciu `zlozena(R1,R2)`, ktorá vytvorí a vráti zloženú reláciu $R = R1 \circ R2$, danú vlastnosťou:

$$(x, z) \in R \Leftrightarrow (\exists y) : (x, y) \in R1 \wedge (y, z) \in R2.$$

Dávajte si pozor na to, aby zoznam, ktorý vaša funkcia bude vracať, neobsahoval viac razy ten istý prvok. Platí aj pre ostatné zadania. Môžete s výhodou použiť typ `set` - trik s vyhadzovaním duplikátov zo zoznamu bol na prednáške.

```
>>> A1=[(1,2),(2,3),(7,3),(10,-1)]
>>> A2=[(2,-4),(2,5),(3,10),(0,-1)]
>>> print zlozena(A1,A2)
[(1, 5), (7, 10), (1, -4), (2, 10)]
```

3. (5 bodov) Vytvorte funkciu `join(R1,R2)`, ktorá vytvorí a vráti ternárnu reláciu (množinu usporiadaných trojíc) $R1 \sqcup R2$, danú predpisom

$$(x, y, z) \in R1 \sqcup R2 \Leftrightarrow (x, y) \in R1 \wedge (x, z) \in R2$$

```
>>> A1=[(1,2),(1,-2),(2,3),(4,0)]
>>> A2=[(0,5),(1,3),(1,-3),(2,100),(5,9)]
>>> join(A1,A2)
[(1, -2, -3), (2, 3, 100), (1, -2, 3), (1, 2, -3), (1, 2, 3)]
```

4. (prémia 5 bodov) Vytvorte funkciu `uzaver(R)`, ktorá vráti najmenšiu možnú reláciu, ktorá je nadmnožinou R a ktorá je tranzitívna.

```
>>> A=[(1,1),(3,4),(3,5),(1,2),(2,3)]
>>> uzaver(A)
[(1, 1), (3, 4), (3, 5), (1, 2), (2, 3), (2, 5), (1, 3), (2, 4), (1, 4), (1, 5)]
```

5. (prémia 5 bodov) Prerobte to, čo ste doteraz spravili, tak, aby to fungovalo objektovo; vytvorte triedu `Relacia`, ktorá bude fungovať takto:

```
>>> A=Relacia([(1,2),(2,3)])
>>> A.symetricka()
False
>>> B=Relacia([(2,5),(2,6)])
>>> C=A.compose(B)
>>> print C
Relacia: (1,5) (1,6)
```

Všimnite si, že aby `print` fungoval tak, ako je na predošlom výpise, musíte implementovať metódu `Relacia.__str__` .