

ZBIERKA ÚLOH - ZADANIA

Úloha 1 Na hodinách matematiky žiaci radi riešia príklady, v ktorých dopĺňajú operátory súčtu, rozdielu, súčinu a podielu medzi dané čísla tak, aby dosiahli daný výsledok, napr. príklad $45_9_1 = 6$ doplnia takto: $45 / 9 + 1 = 6$.

Vytvorte pre pani učiteľku matematiky program **uprava.py**, ktorému na vstupe zadá vyriešený príklad uvedeného typu v tvare `číslo operátor číslo operátor číslo ... = číslo` a nahradí v ňom všetky operátory znakom `_`.

Napr. po zadaní reťazca `45 / 9 + 1 = 6` program vypíše upravený reťazec `45 _ 9 _ 1 = 6`.

Úloha 2 Vo výskumnom oddelení firmy **Water Locator** vyvíjajú nový typ robota, ktorý vyhľadáva pomocou špeciálnych senzorov zdroj podzemnej vody (a pomáha tak určiť miesto, kde kopať studňu). Žiaľ, nemajú ešte zvládnutý algoritmus pohybu robota po pozemku – pohybuje sa rovnako dlhými krokmi, pričom pred každým krokom si náhodne zvolí jeden zo štyroch smerov: sever (S), juh (J), západ (Z) a východ (V). Kód aktuálneho smeru pohybu ukladá robot do reťazca, napr. reťazec 'SSVSJJZ' reprezentuje pohyb sever – sever – východ – sever – juh – juh – západ. Keď robot nájde podzemný vodný zdroj, ukončí zápis do reťazca a vráti sa na svoju štartovaciu pozíciu.

Problémom je, že reťazec reprezentujúci pohyb robota je často krát dlhý a je neefektívne prechádzať celú trasu podľa reťazca. Preto je našou úlohou definovať funkciu, ktorá upraví reťazec tak, že bude určovať najkratšiu cestu zo štartovacej pozície k nájdenému podzemnému zdroju.

ZBIERKA ÚLOH – RIEŠENIA

Úloha 3 Na hodinách matematiky žiaci radi riešia príklady, v ktorých dopĺňajú operátory súčtu, rozdielu, súčinu a podielu medzi dané čísla tak, aby dosiahli daný výsledok, napr. príklad $45_9_1 = 6$ doplnia takto: $45 / 9 + 1 = 6$.

Vytvorte pre pani učiteľku matematiky program **uprava.py**, ktorému na vstupe zadá vyriešený príklad uvedeného typu v tvare číslo operátor číslo operátor číslo ... = číslo a nahradí v ňom všetky operátory znakom `_`.

Napr. po zadaní reťazca $45 / 9 + 1 = 6$ program vypíše upravený reťazec $45_9_1 = 6$.

Riešenie:

```
text = input('Zadajte matematický príklad: ')
nahrad = '+-*/'
for operator in nahrad:
    text = text.replace(operator, '_')
print(text)
```

Úloha 2 Vo výskumnom oddelení firmy **Water Locator** vyvíjajú nový typ robota, ktorý vyhľadáva pomocou špeciálnych senzorov zdroj podzemnej vody (a pomáha tak určiť miesto, kde kopať studňu). Žiaľ, nemajú ešte zvládnutý algoritmus pohybu robota po pozemku – pohybuje sa rovnako dlhými krokmi, pričom pred každým krokom si náhodne zvolí jeden zo štyroch smerov: sever (S), juh (J), západ (Z) a východ (V). Kód aktuálneho smeru pohybu ukladá robot do reťazca, napr. reťazec 'SSVSJJZ' reprezentuje pohyb sever – sever – východ – sever – juh – juh – západ. Keď robot nájde podzemný vodný zdroj, ukončí zápis do reťazca a vráti sa na svoju štartovaciu pozíciu.

Problémom je, že reťazec reprezentujúci pohyb robota je často krát dlhý a je neefektívne prechádzať celú trasu podľa reťazca. Preto je našou úlohou definovať funkciu, ktorá upraví reťazec tak, že bude určovať najkratšiu cestu zo štartovacej pozície k nájdenému podzemnému zdroju.

Riešenie:

```
def skratka(smer1, smer2, rozdiel):
    if rozdiel > 0:
        smer = smer1
    elif rozdiel < 0:
        smer = smer2
    else:
        smer = ''

    upravena_cesta = ''
    for i in range(abs(rozdiel)):
        upravena_cesta = upravena_cesta + smer
    return upravena_cesta

cesta = 'SSSSJJVVZ'
cesta = cesta.upper()

sever_juh = cesta.count('S') - cesta.count('J')
vychod_zapad = cesta.count('V') - cesta.count('Z')

kratka_cesta = skratka('S', 'J', sever_juh)
kratka_cesta = kratka_cesta + skratka('V', 'Z', vychod_zapad)

print(kratka_cesta)
```