PRACOVNÝ LIST

PROGRAMOVANIE – Zoznam, vyhľadávanie, krokovanie

Úloha 1

V konzolovom režime vytvorte pracovný zoznam **ovocie**:

**ovocie = [' pomaranč ',** **'jablko',** **'banán',** **'kiwi', 'banán',** **'hruška', 'kiwi', 'melón']**

Najprv odhadnite, čo bude výsledkom nasledujúcich príkazov (ak neviete, uveďte NEVIEM alebo otáznik), potom si ich odskúšajte a skontrolujte správnosť svojich predpokladov:

|  |  |  |
| --- | --- | --- |
| **príkaz** | **výsledok (odhad)** | **výsledok (overenie)** |
| **'**melón**'** in ovocie |  |  |
| **'**marhuľa**'** in ovocie |  |  |
| ovocie[1] |  |  |
| ovocie[2] |  |  |
| ovocie[8] |  |  |
| ovocie[-1] |  |  |
| ovocie.index(**'**jablko**'**) |  |  |
| ovocie.index(**'**pomaranč**'**) |  |  |
| ovocie.index(**'**marhuľa**'**) |  |  |
| ovocie.index(**'**kiwi**'**) |  |  |
| ovocie.count(**'**jablko**'**) |  |  |
| ovocie.count(**'**banán**'**) |  |  |
| len(ovocie) |  |  |

Úloha 2

V konzolovom režime vytvorte pracovný zoznam **dni**:

**dni = ['pondelok',** **'utorok',** **'streda',** **'štvrtok', 'piatok',** **'sobota',** **'nedeľa']**

Vyskúšajte, čo bude výsledkom nasledujúcich operácií a navrhnite vhodné názvy pre takto vzniknuté podzoznamy:

|  |  |  |
| --- | --- | --- |
| **príkaz** | **výsledok** | **vhodný názov podzoznamu** |
| dni[:5] |  |  |
| dni[5:] |  |  |

Doplňte, ako by ste pomocou rezov vytvorili nasledujúce podzoznamy:

**['utorok',** **'štvrtok', 'sobota'] → dni[ ]**

**['pondelok',** **'streda',** **'piatok'] → dni[ ]**

**['pondelok',** **'streda',** **'piatok',** **'nedeľa'] → dni[ ]**

Úloha 3

V konzolovom režime vytvorte pracovný zoznam **dni**:

**dni = ['pondelok',** **'utorok',** **'streda',** **'štvrtok', 'piatok',** **'sobota',** **'nedeľa']**

Otestujte nasledujúce príkazy – použite zakaždým **kontrolný výpis** pomocou príkazu **print(dni)**:

|  |  |
| --- | --- |
| **príkaz** | **výsledok** |
| dni.remove(**'**streda**'**) |  |
| dni.append(**'**streda**'**) |  |
| dni.insert(2, **'**streda**'**) |  |

Aký je rozdiel medzi použitím **append** a **insert**?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Na čo slúži **remove**?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Úloha 4

V konzolovom režime vytvorte pracovný zoznam **dni**:

**dni = ['pondelok',** **'utorok',** **'streda',** **'štvrtok', 'piatok',** **'sobota',** **'nedeľa']**

Otestujte nasledujúce príkazy – použite zakaždým **kontrolný výpis** pomocou príkazu **print(dni)**:

|  |  |
| --- | --- |
| **príkaz** | **výsledok** |
| dni.reverse() |  |
| dni.sort() |  |

Úloha 5 (povinná)

Vytvorte program **pismeno.py**, ktorý načíta od používateľa písmeno a určí, či je samohláskou alebo spoluhláskou. Pri riešení použite zoznam.

Úloha 6 (povinná)

Vytvorte program **kalendar.py**, ktorý načíta kalendárny mesiac a určí, či má 31, 30 alebo menej ako 30 dní.

Úloha 7 (povinná)

Doplňte program **priezvisko.py**, aby prerozdelil priezviská zo zoznamu **hostia** do zoznamov **muzi** a **zeny.** Potom postupne vypíšte abecedne usporiadaných všetkých mužov a všetky ženy.

Úloha 8 (povinná)

Meteostanica meria denné teploty o 7:00, 14:00 a 21:00. Priemernú dennú teplotu vypočítame podľa vzťahu tdenná=(t7:00 + t14:00 + 2 \* t21:00)/4 . Vytvorte program **teplota.py**, ktorý bude postupne načítavať teploty v priebehu dňa a ukladať ich do zoznamu. Na koniec zoznamu pridá napokon vypočítanú priemernú dennú teplotu a vypíše ju.

Úloha 9 (povinná)

V programe **obraty.py** sú zaznamenané jednotlivé obraty na účte v banke (vklady a výbery). Doplňte program, aby vypísal, koľko bolo na účet celkovo vložených prostriedkov, koľko celkovo bolo z účtu čerpaných prostriedkov a aký je výsledný zostatok na účte po danom sledovanom období.

Úloha 10 (povinná)

Doplňte v programe **vycisti.py** funkciu **cisti\_zoznam()** tak, aby načítala vstupný zoznam miest a vrátila zoznam, ktorý neobsahuje duplicity (t.j. každé mesto sa v ňom bude vyskytovať len raz).

Úloha 11 (dobrovoľná)

Doplňte program **robot.py**, aby vykresľoval pohyb robota v skladisku na základe inštrukcií zo zoznamu **pohyb**. Robot sa dokáže len pohnúť dopredu o 10 krokov (**D**) alebo otočiť vpravo o 90° (**T**).

Úloha 12 (sebahodnotiaci test)

Aký bude konečný výsledok nasledujúcich dvoch programových kódov?

**PROGRAM A:**

mesta=[**'**Košice**'**, **'**Bardejov**'**, **'**Martin**'**, **'**Poprad**'**, **'**Martin**'**, **'**Trenčín**'**, **'**Prešov**'**, **'**Košice**'**,

**'**Martin**'**]

mesta.sort()

mesta.reverse()

mesta[**4**]

**VÝSLEDOK/ODPOVEĎ:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROGRAM B:**

mesta=[**'**Košice**'**, **'**Bardejov**'**, **'**Martin**'**, **'**Poprad**'**, **'**Martin**'**, **'**Trenčín**'**, **'**Prešov**'**, **'**Košice**'**,

**'**Martin**'**]

novy=[]

**for** i **in** mesta:

**if** i != "Košice":

novy.append(i)

novy[-**2**]

**VÝSLEDOK/ODPOVEĎ:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_