

## Zadání 1 – opakování (procvičení) programování v Pythonu

1. Navrhněte malou, vhodně reprezentovanou databázi domácích mazlíčků a určete relace mezi termy. Nad databází definujte množinu zvířat a ověřte platnost Vámi definované množiny mazlíčků. Nad databází si vyzkoušejte zadávání cílů a prověřte vyhodnocování cílů. Úlohu realizujte v Pythonu.
2. V Pythonu realizujte úlohu jednoduchého početního výukového nebo herního programu. Jednotlivé početní operace volte podle uvážení.
3. Zformulujte program v Pythonu, kterým definujete množinu odpovědí na různé typy pozdravů. Volte alespoň pět různých zadávaných pozdravů a programem realizujte všechny možnosti slušných odpovědí na pozdrav. Na neporozuměný dotaz necht' program odpovídá slušně, odpovídajícím způsobem.
4. Definujte množinu min. 5 mužů a 5 žen a mezi nimi všechny možné stavy (svobodní, manželé, milenci, ovdovění, rozvedení – stavy vhodně konkrétním osobám pevně zadejte) a zapište program v Pythonu, kterým na dotaz zjistíte, které dvojice jsou manželé, které milenci, kdo žije sám atp.
5. Zformulujte program v Pythonu, kterým realizujete program typu jednoduchého kvízu, jímž budete u uživatele prověřovat znalost jednoduchých aritmetických operací.
6. Zformulujte program v Pythonu, kterým realizujete program typu jednoduchého kvízu, jímž budete u uživatele prověřovat znalost logických operací (minimálně not, and, or, xor).
7. Realizujte program v Pythonu, kterým budete demonstrovat funkci (vyhodnocování) symetrických a nesymetrických relací; zvolte k tomu vhodnou množinu dat a podmínky vyhodnocování.
8. Zapište v Pythonu jednoduchý postup, kterým přečtete všechny termy ze souboru A a uspořádaně (relaci uspořádání definujte) je pak zapište do souboru B. Programově realizujte a funkci programu ověřte jak na číselných, tak na alfanumerických datech.
9. Realizujte jednoduchý program v Pythonu, kterým najdete nejkratší, nejlevnější, nejrychlejší... cestu v orientovaném ohodnoceném grafu s alespoň deseti uzly.
10. Realizujte program v Pythonu, jímž vytvoříte jednoduchou neuspořádanou tabulku o třech sloupcích [číselný údaj, alfanumerický údaj (text), číselný údaj] a obsah této tabulky budete na přání uživatele vypisovat vzestupně uspořádanou podle hodnot (klíčů) v prvním nebo druhém sloupci tabulky.
11. Realizujte program v Pythonu, jímž vytvoříte jednoduchou seznamovou strukturu (pole tříprvkových seznamů obsahujících číselný údaj, alfanumerický údaj (text), číselný údaj) a obsah tohoto pole seznamů budete na přání uživatele vypisovat vzestupně uspořádaný podle hodnot (klíčů) uložených v prvním nebo druhém prvku seznamu.

12. Zapište program v Pythonu, kterým implementujete operace výpočtu faktoriálu, druhé a třetí mocniny zadaného údaje, nalezení největšího společného dělitele, zjištění prvočíselnosti, zbytku při dělení apod.
13. Realizujte program v Pythonu, kterým pro zadané aritmetické výrazy otestujete, zda jsou zapsány v prefixové, infixové nebo postfixové formě (formátu).
14. Realizujte program v Pythonu, kterým budete vypisovat požadovaný (zadaný) počet členů číselných, funkčních, mocninných, Taylorových, Fourierových a Fibonacciho řad (posloupností).
15. Realizujte program v Pythonu, jímž budete simulovat efektivní přesouvání kotoučů hanojské věže ze zadaného zdrojového kolíku (kde se nachází věž) na zadaný cílový kolík pro vyšší počet kotoučů (max. počet kotoučů budiž  $\geq 20$ , mezní hodnota = 64).
16. Realizujte program v Pythonu, jímž budete modelovat a na obrazovce dokumentovat provádění základních množinových operací; typy a velikost množin volte podle uvážení
17. Naprogramujte želvu, aby kreslila pěkné obrázky. Želva rozumí pokynům jako je „jdi dopředu“ či „zatoč doprava“. Pro rozcvičení můžete použít verzi želví grafiky v grafickém programování. Želvu dále můžete učit nové příkazy, např. diamant, kytičky, maltézský kříž apod., – na tom si dobře procvičíte funkce v Pythonu a navíc vykreslíte zajímavé obrázky.
18. Programátorská úkolovka – v úloze je úkolem určit tajný kód pomocí výpočtu. Jak výpočet provedete, záleží jen na vás, jako odpověď zadáváte pouze výsledný kód. Úloha je formulována tak, aby byla snadno řešitelná v programovacím jazyku Python.
19. Zpracujte program v Pythonu, který vyčíslí součet všech prvočísel, která jsou menší než 1000 a neobsahují číslici 3. Vypište jak vyčíslený součet, tak posloupnost sčítaných čísel.
20. Číselná morseovka: Morseova abeceda se zapisuje pomocí čárek, teček a oddělovačů. V následující zprávě jsou oddělovače zapsány jako nuly, tečky jsou ostatní sudé číslice, čárky jsou liché číslice:

9748045074804045190708050630237505608086260406860838207710561019309507  
242021083950120

Překlad jednotlivých písmen na morseovku uvádí následující slovník:

```
morse = {"A": ".-.", "B": "-...", "C": "-.-.", "D": "-..", "E": ".",
         "F": "..-.", "G": "--.", "H": "....", "I": "..", "J": ".---",
         "K": "-.-.", "L": ".-..", "M": "--", "N": "-.", "O": "---",
         "P": ".-.-.", "Q": "--.-", "R": "-.-.", "S": "...", "T": "-.",
         "U": "..-.", "V": "...-", "W": ".-.-", "X": "-.-.-", "Y": "-.-.",
         "Z": "--.."}

```

Zpracujte program v Pythonu, který uvedenou transformaci realizuje a výsledek vypíše.

21. Fibonacciho posloupnost má jako první dva členy jedničky, každý další člen je součtem dvou předchozích. Posloupnost tedy začíná 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144 ... . Vás budou zajímat pouze sudé členy této posloupnosti – například čtvrté sudé Fibonacciho číslo je 144. Programem v Pythonu určete, jaká bude hodnota zvoleného sudého Fibonacciho čísla (pořadí zvoleného sudého čísla zadávejte, např. čtrnácté).

22. Hledáme čísla, která mají co nejvíce přirozených dělitelů. Z čísel do 50 je to číslo 48, které má 10 dělitelů. Napište program v Pythonu, kterým zjistíte, které z čísel menších než 1000 má nejvíce dělitelů.
23. Některá čísla lze vyjádřit jako součet tří druhých mocnin přirozených čísel (tedy celých čísel větších jak nula). Například  $964 = 6 * 6 + 12 * 12 + 28 * 28$ . Jiná čísla takto vyjádřit nelze, například číslo 7. Zpracujte program v Pythonu, který zjistí a vypíše, kolik přirozených čísel od 1 do 1000 (včetně) lze vyjádřit jako součet tří druhých mocnin přirozených čísel?