**Úkoly ke cvičení z 14.3.**

**Úkol 1:** Od uživatele získejte vstupní hodnoty, které vymezují interval, v němž budete hledat řešení. Oba krajní body do tohoto intervalu budou patřit. Najděte všechny čísla celočíselně bezezbytku dělitelná 3 a současně i 8. A pro ty to hodnoty vytiskněte: *Číslo x je dělitelné bezezbytku 3 i 8.* Kde x je konkrétní číslo.

**Úkol 2:** Od uživatele získejte dvě vstupní hodnoty, kde první určuje číslo, které se bude umocňovat. Zjistěte, na kterou mocninu je třeba umocnit první vstupní hodnotu, aby byla striktně větší než druhé zadané číslo.

*Příklad: vstupy 2 a 9. 24(=16)>9, naproti tomu 23(=8)<9, takže výsledkem je 4.*

**Úkol 3:** Projděte interval čísel od 1000 do 450 s krokem -10 a zjistěte počet čísel dělitelných celočíselně číslem 16 ze zbytkem právě 8. Netiskněte čísla, pouze na závěr jejich počet.

**Úkol 4:** Vytiskněte 15 řádků, na každém řádku bude právě 15 znaků. První řádek bude začínat jednou \* a bude následovat čtrnáct /, na každém řádku pak přibude jedna \* a ubyde jedno /. Poslední řádek budou pouze \*.

*Příklad:*

*\*//////////////*

*\*\*/////////////*

*Atd.*

**Úkol 5:** Zjistěte zda-li v intervalu zadaném od uživatele lze najít číslo, které je dělitelné bezezbytku 17 a zároveň i 14. Pokud ano, vypište, že číslo bylo nalezeno. Pokud ne, pokuste se takové číslo najít v intervalu, který bude rozšířený na obě strany o 50. Pokud tam hodnota existuje, vypište, že číslo lze najít v rozšířeném intervalu. Pokud ani tam řešení není, pak vypište, že hodnotu nelze najít.

**Úkoly ke cvičení z 21.3.**

**Úkol 1:** Napište funkci, která jako vstup přijímá dvě proměnné. Předpokládejte, že vstupy jsou seznamy. Porovnejte jejich délku (funkce len(listName)), pokud obsahují stejný počet prvků funkce vrací hodnotu true, jinak vrací hodnotu false.

**Úkol 2:** Napište funkci, která jako argument přímá počet prvků v seznamu - *n*. Funkce od uživatele bude požadovat *n* vstupů, ze kterých vytvoří seznam, který bude funkce vracet.

**Úkol 3:** Napište funkci, která jako argument přijímá seznam hodnot. Předpokládejte, že seznam obsahuje pouze čísla. Funkce vrací jako výsledek průměr z čísel v seznamu.

**Úkol 4:** Napište funkci, která ze vstupního řetězce vrátí znaky v rozmezí 5 - 9 znaku. Pokud je řetězec kratší než 9 znaků, vypište hlášku, že operaci nelze provést. (dělení řetězců zde - <http://stackoverflow.com/questions/663171/is-there-a-way-to-substring-a-string-in-python>)

**Úkol 5:** Napište funkci, která jako argument přijímá řetězec. Upravte vstupní řetězec tak, aby obsahoval pouze malá písmena. Následně nahraďte všechny české znaky (s háčky, čárkami a kroužky) v řetězci za jejich ekvivalenty bez těchto znaků. Takto upravený řetězec bude návratovou hodnotou funkce.

Při řešení se zkuste zamyslet nad logickým a úsporným uspořádáním kódu. V rámci kódu výsledné funkce by se žádné řádky neměly, byť drobně upravené, opakovat. Celou výslednou funkci lze zapsat i do 7 řádků kódu. Vaše řešení by se nemělo délkou příliš lišit.

*Příklad:* vstup - Příšerně žluťoučký kůň úpěl ďábelské ódy.

výstup - priserne zlutoucky kun upel dabelske ody.

*Poznámka:* Správnost a kompletnost řešení si můžete ověřit na českých Pangramech (<http://cs.wikipedia.org/wiki/Pangram>).

**Úkoly ke cvičení z 28.3.**

**Některé úkoly jsou založené na práci se soubory, které najdete ke stažení ve složce pod názvem: data\_ke\_cviceni\_z\_28.3.zip.**

**Úkol 1:** Napište program, který načte soubor *ukol\_1\_data.txt*. V souboru je *n* celých čísel z intervalu 1 – 1 000 000. Zjistěte počet čísel větších než 9 00 000 a sumu čísel menších než 1 000. Obě hodnoty vytiskněte na obrazovku.

**Úkol 2:** Napište program, který načte soubor *ukol\_2\_data.txt*. Zkontrolujte si tento soubor. V souboru je *n* řádků obsahujících dvě desetinná čísla oddělená oddělovačem. Pro každý řádek zjistěte rozdíl cislo1 – cislo2. Spočte počet a celkovou sumu pro situace kdy je tento rozdíl kladný a zvlášť stejné charakteristiky pro záporný rozdíl. Na konci programu vytiskněte vhodné textové zhodnocení výsledků pro uživatele.

**Úkol 3:** Napište program, který načte soubor *ukol\_3\_data.txt*. Zkontrolujte si tento soubor. Soubor obsahuje *n* řetězců o 9 znacích oddělené oddělovačem. Vyberte z těchto řetězců všechny obsahující podřetězec „99“ (bez uvozovek) a uložte je do pole. Nakonec pole vytiskněte.

**Úkol 4:** Napište program, který načte soubor *ukol\_4\_data.txt*. Zkontrolujte si tento soubor. Soubor obsahuje *n* řádků s hodnotami *x,y* a *z* oddělené oddělovačem. Pro každý řádek spočítejte funkci *sin(x) + cos(y) – log10(z)*. Výsledky ukládejte do pole a pole na závěr vytiskněte.

**Úkol 5:** Napište program, který načte soubor *ukol\_5\_data.txt*. Zkontrolujte si tento soubor. Soubor obsahuje *n* řádků s hodnotou. Pro každý řádek zjistěte hodnotu nejmenšího celého čísla většího nebo rovno než hodnota na řádku, a hodnotu největšího celého čísla menšího nebo rovno hodnotě na řádku. Výsledné hodnoty zapište do souboru *ukol\_5\_vysledek.txt* ve formátu *mensi;vetsi.* Každému řádku původního soboru odpovídá řádek nového souboru.

**Úkoly ke cvičení ze 4.4.**

**Některé úkoly jsou založené na práci se soubory, které najdete ke stažení ve složce pod názvem: data\_ke\_cviceni\_ze\_4.4.zip.**

**Úkol 1:** Napište program, který vygeneruje soubor *ukol\_1\_data.txt*. Soubor bude obsahovat hlavičku *uniform;gauss;lognormvariate* a následně 1 000 řádků, s náhodnými čísly vygenerovanými ze zadaných rozdělení (parametry rozdělení jsou na vás, nepožadují se žádné konkrétní). Oddělovačem hodnot bude středník.

**Úkol 2:** Napište program, který načte soubor *ukol\_2\_data.txt*. Z datasetu vyseparujte do seznamu čtvrtý sloupec hodnot. Tento seznam vytiskněte do souboru *ukol\_2\_4\_sloupec.txt* tak, že na každém řádku bude právě jedna hodnota.

**Úkol 3:** Napište funkci, která má tři argumenty – první je název souboru v textové podobě, druhý parametr je booleovská hodnota a třetí je index. Funkce načte soubor, pokud je booleovská hodnota *True*, pak načte data z řádku definovaného třetím argumentem. Pokud je tato hodnota *False*, pak funkce načte sloupec s příslušným indexem. Funkce v obou případech vrací seznam čísel s plovoucí desetinnou čárkou.

*Poznámka:* Oddělovačem čísel v souboru bude vždy mezera. Funkci můžete otestovat na souboru z druhého úkolu. Dejte pozor na ukončování řádku a jeho chování! V některém případě bude možná zapotřebí tento znak manuálně odstranit

**Úkol 4:** Ze souboru v **Úkolu 3** si naimportujte vytvořenou funkci. Funkci zavolejte s parametry *("ukol\_2\_data.txt", False, 55)* . Seznam hodnot vytiskněte na obrazovku. V seznamu, který funkce vrátí zjistěte nejmenší a největší hodnotu a jejich indexy v rámci seznamu.

**Úkol 5:** Ze souboru v **Úkolu 3** si naimportujte vytvořenou funkci. Funkci zavolejte s parametry *("ukol\_2\_data.txt", True, 100)*. V seznamu, který funkce vrátí zjistěte počet hodnot 0 a 1, dále počet hodnot větších než 0.5 ale nerovnajících se 1 a všech ostatních, které nevyhovují ani jedné z těchto tří podmínek. Pro každý počet hodnot vytiskněte informaci: *Počet hodnot x je y což je z % z celkového počtu.*