

## Cvicenie 5

### Instrukcie:

- Vypracujte vsetky ulohy. Na cviceni sa pokuste vypracovat co najviac uloh a ulohy, ktore nestihnute na cviceni, potom vypracujte doma.
- **Pozor! Nevytvarajte rekurzivne funkcie pomocou globalnych premennych!** Je to zly zvyk a na skuske bude za taketo riesenia 0 bodov!
- **Specialne davam do pozornosti ulohu cislo 7, v ktorej sa oboznamite so slavnou a podla mna aj velmi peknu algoritmicou ulohou „Towers of Hanoi“.** Ak sa vam ulohu 7 na cviceni podari vyriesit, ohlaste sa cviciacemu, odprezentujte mu vase riesenie a ak to budete mat dobre, dostanete jeden **bonusovy bod**.
- **V pripade, ze sa na niektorej ulohe zaseknete, pytajte sa cviciaceho.**

### Cast prva: Debugger

1. Vyvojove prostredia pre tvorbu pocitacovych programov spravidla obsahuju nastroj zvany Debugger. Pomocou tohto nastroja mozete prechadzat vasim programom krok po kroku a mozete sledovat, ako sa menia hodnoty premennych vo vasom programe. Toto je velmi uzitocne pri hladani chyb vo vasom programe. Prejdite si tutorial o pouzivani Debuggera vo vyvojovom prostredi IDLE dostupny na:

<https://www.cs.uky.edu/~keen/help/debug-tutorial/debug.html>

Osvojte si pracu s Debuggerom v IDLE. **Bude to pre vas uzitocne!** Ak radsej pouzivajte ine vyvojove prostredie ako IDLE, osvojte si pracu s Debuggerom vo vasom preferovanom vyvojovom prostredi!

### Cast druha: Rekurzia

1. Definujte funkciu s parametrom  $n$ , ktora vrati sucet  $1+2+3+\dots+n$ . Vo funkcii pouzite **rekurziu!**
2. Definujte funkciu s parametrom  $n$ . Funkcia nacita z klavesnice  $n$  cisel a vrati sucet nacitanych cisel. Vo funkcii pouzite **rekurziu!**
3. Definujte funkciu s parametrom  $n$ . Funkcia nacita z klavesnice  $n$  cisel a vrati pocet, kolko z nacitanych cisel bolo parnych. Vo funkcii pouzite **rekurziu!**
4. Definujte funkciu s parametrom  $n$ . Funkcia nacita z klavesnice  $n$  cisel a vrati najvacsie z nacitanych cisel. Vo funkcii pouzite **rekurziu!**
5. Definujte funkciu s parametrom  $n$ . Funkcia nacita z klavesnice  $n$  cisel a vrati hodnotu True, ak je sucet nacitanych cisel parny. V opacnom pripade vrati funkcia hodnotu False. Vo funkcii pouzite **rekurziu!**  
*Pomocka:* Zamyslite sa, ako sa zmeni parita cisla, ak k nemu pridame parne cislo, a ako sa zmeni, ak k nemu pridame neparne cislo.
6. Definujte funkciu, ktora pre argument  $n$  vrati sucet prvocisel mensich ako  $n$ . Vo funkcii pouzite **rekurziu!** Mozete predpokladat, ze mate k dispozicii funkciu *test\_prvociselnosti* z minuleho cvicenia.

7. **Bonusova uloha.** Vsetky vyssie uvedene ulohy by sa dali jednoducho vyriesit aj bez rekurzie. Rekurzia nam ale niekedy umoznuje jednoducho riesit aj ulohy, ktore by sa nam inak riesili velmi tazko. Prikladom takejto ulohy je slavná algoritmicke uloha “Towers of Hanoi”.

a) Pozrite si toto video o ulohe “Towers of Hanoi”:

<https://www.youtube.com/watch?v=0u7g9C0wSIA>

Podľa mňa je veľmi pekne a zaujímavé :)

Poznámka: V skripte vo videu autor používa funkciu `str()`. Je to funkcia, pomocou ktorej pretypujeme objekt na reťazec (reťazec=postupnosť znakov, po anglicky string). Podobne funguje funkcia `int()`, s ktorou ste sa už stretli, funkcia `int()` ale pretypováva objekt na celé číslo.

b) V skripte vo videu vyššie je funkcia `Towers(n, fr, to, spare)`, ktorá vypíše sadu inštrukcií ako vyriešiť úlohu “Towers of Hanoi”, ak na začiatku máme `n` diskov na stĺpe s názvom `fr` a tieto disky chceme presunúť na stĺp s názvom `to` a máme ešte k dispozícii stĺp s názvom `spare`. Vasou úlohou je teraz definovať funkciu `Towers2(n)`, ktorá vráti, koľko presunov diskov treba vykonať, ak chceme vyriešiť úlohu “Towers of Hanoi” s `n` diskami (inak povedané funkcia `Towers2(n)` vráti počet riadkov, ktoré vypíše funkcia `Towers(n, fr, to, spare)`). Funkcia `Towers2(n)` musí byť rekurzívna, nesmie volať funkciu `Towers(n, fr, to, spare)` a nesmie nič vypisovať.

8. Vyriešte cvičenie 6.3 na strane 61 v knihe.

9. Vyriešte cvičenie 6.4 na strane 61 v knihe.

10. Vyriešte cvičenie 6.5 na strane 61 v knihe.

11. Vyriešte cvičenie 5.6 na strane 49 v knihe.