

IB113 Úvod do programování a algoritmizace

Cvičení 7

Jaromír Plhák



Domácí úkol 2

- Pep8
 - 5 byla má hranice – příště to budou pouze 3 atd.
 - Recidiva se nevyplácí
- Vyhýbat se použití jednopísmenným proměnným
- Problém typu if: return else:
 - Zatím jsem nepenalizoval, příště budu
- Opakování kódu
 - Špatné rozmyšlení problému
 - Nevyužití již implementovaných funkcí

Domácí úkol 2

- Porušení pravidel
 - Házení jinak než podle pravidel
 - Přeskočení domečku
- Porušení zadání
 - Změna hlaviček funkcí
- Dlouhé funkce
- while True:
- if <podmínka>: pass else:

Domácí úkol 3

- Zadáno v Isu
 - https://is.muni.cz/auth/el/1433/podzim2017/IB113/um/homework_03/
- Je na vás, jaké úkoly implementujete
 - 5 úkolů z easy
 - 2 body za vyřešení
 - 2 úkoly z advanced
 - 5 bodů za vyřešení
- Zadání v angličtině
- V případě nějakých problémů, pište na diskuzní fórum
- Odevzdávejte archiv (.zip, .rar) obsahující všechna řešení
- Deadline čtvrtek 9. 11. 23:59

Úkoly

- 6.1.1. Hádání čísla člověkem ¶

- Napište funkci `guess_number_human(upper_bound)`, která umožňuje hrát s počítačem hru na hádání čísla: počítač si myslí číslo (celé číslo v intervalu $[1, upper_bound]$), hráč se ho snaží uhodnout. Po každém pokusu dostane hráč od počítače informaci, zda je hledané číslo menší nebo větší než to, které si tipnul. Hru si můžete vyzkoušet zde (klikejte na čísla v pravé části).

- 6.1.2. Hádání čísla počítačem

- Napište funkci `guess_number_pc(upper_bound)`, která umožňuje hrát s počítačem hru na hádání čísla, tentokrát si však číslo myslí uživatel a počítač hádá. Po každém pokusu si počítač vyžádá od uživatele informaci, zda je myšlené číslo větší nebo menší než to, které si počítač tipnul.

Úkoly

- 6.2.1. Binární vyhledávání

- Napište funkci `binary_search(needle, haystack)`, která zjistí, zda se hodnota `needle` nachází ve vzestupně uspořádaném seznamu `haystack`. Funkce musí mít logaritmickou časovou složitost.