

Cvičenie 4

Inštrukcie:

- Vypracujte všetky úlohy. Na cvičení sa pokúste vypracovať čo najviac úloh a úlohy, ktoré nestihnete na cvičení, potom vypracujte doma.
- **V prípade, že sa na niektorej úlohe zaseknete, pýtajte sa cvičiacich!**
- **Ak to nie je uvedené v zadaní, nemusíte ošetrovať platnosť vstupov/parametrov.**

Sekcia 1:

Úloha č. 1

Definujte funkciu *test_parity(n)* s parametrom *n*, ktorá **vráti** hodnotu *True*, ak *n* je párne číslo a vráti hodnotu *False*, ak *n* je nepárne číslo.

Ukážky vstupov a výstupov:

Volanie *test_parity(2)* vráti hodnotu *True*

Volanie *test_parity(0)* vráti hodnotu *True*

Volanie *test_parity(5)* vráti hodnotu *False*

Volanie *test_parity(13)* vráti hodnotu *False*

Úloha č. 2

Definujte funkciu *minimum_dvoch(a,b)* s 2 parametrami, číslami *a* a *b*. Funkcia **vráti** minimum z týchto 2 čísiel.

Ukážky vstupov a výstupov:

Volanie *minimum_dvoch(2,5)* vráti hodnotu 2

Volanie *minimum_dvoch(5,5)* vráti hodnotu 5

Volanie *minimum_dvoch(-2,-105)* vráti hodnotu -105

Úloha č. 3

Definujte funkciu *minimum_troch(a,b,c)* s 3 parametrami, číslami *a,b,c*. Funkcia **vráti** minimum z týchto 3 čísiel.

Ukážky vstupov a výstupov:

Volanie *minimum_troch(2,5,-2)* vráti hodnotu -2

Volanie *minimum_troch(5,1,10)* vráti hodnotu 1

Volanie *minimum_troch(-105,-105,1)* vráti hodnotu -105

Úloha č. 4

Definujte funkciu *menu()* bez parametrov. Funkcia načíta vstup od používateľa z klávesnice. V prípade, že používateľ zadal znak *s*, funkcia pomocou modulu *turtle* vykreslí štvorec so stranou dĺžky 100 pixelov; v prípade, že používateľ zadal znak *t*, funkcia pomocou modulu *turtle* vykreslí rovnostranný trojuholník so stranou dĺžky 100 pixelov; v prípade, že používateľ zadal čokoľvek iné, funkcia vypíše na obrazovku správu *Zadali ste neplatný vstup*

Úloha č. 5

Definujte funkciu *pocet_rovnakych(a,b,c)* s 3 parametrami, číslami *a,b,c*. Funkcia **vráti** číslo 3, ak sú všetky tri čísla *a,b,c* rovnaké; vráti číslo 2 ak sú dve z čísel *a,b,c* rovnaké a tretie číslo je iné, a vráti číslo 0, ak sú všetky tri čísla rôzne.

Ukážky vstupov a výstupov:

Volanie *pocet_rovnakych(2,5,8)* **vráti** hodnotu 0; volanie *pocet_rovnakych(1,2,1)* **vráti** 2; volanie *pocet_rovnakych(10,10,10)* **vráti** 3.

Úloha č. 6

Definujte funkciu *pocet_delitelnych_5(n)*, ktorá má vstupný parameter celé nezáporné číslo n . Funkcia načíta n čísiel z klávesnice a **vráti** počet načítaných čísiel, ktoré boli deliteľné číslom 5. Ošetríte situáciu, ak by bola funkcia volaná so záporným vstupom! V takom prípade nech funkcia vypíše, že vstupný argument musí byť nezáporné číslo a vráti hodnotu -1.

Ukážky vstupov a výstupov:

Volanie *pocet_delitelnych_5(5)* načíta 5 čísiel z klávesnice. Ak čísla sú napríklad 5,10,3,4,0 funkcia **vráti** číslo 3 (pretože 5, 10 a 0 sú deliteľné 5).

Volanie *pocet_delitelnych_5(4)* načíta 4 čísla z klávesnice. Ak čísla sú napríklad 1,2,3,4 funkcia **vráti** číslo 0 (pretože ani jedno z čísiel nie je deliteľné 5).

Volanie *pocet_delitelnych_5(-3)* vypíše, že vstupný argument musí byť nezáporné číslo a **vráti** číslo -1.

Úloha č. 7

Definujte všeobecnejšiu verziu funkciu z predošlej úlohy, *pocet_delitelnych(n,k)*. Funkcia má 2 vstupné parametre, nezáporné celé číslo n a kladné číslo k . Ošetríte situáciu, že by vstupný argument pre parameter n bolo záporné číslo alebo že by vstupný argument pre parameter k bolo záporné číslo alebo nula – vždy vypíšete príslušnú chybovú hlášku a funkcia vráti hodnotu -1. V prípade, že sú vstupné argumenty v poriadku, funkcia načíta n čísiel a **vráti** počet načítaných čísiel, ktoré boli deliteľné číslom k .

Ukážky vstupov a výstupov:

Volanie *pocet_delitelnych(5,2)* načíta 5 čísiel z klávesnice a vráti koľko z nich je deliteľných 2. Ak čísla sú napríklad 5,10,3,4,0 funkcia **vráti** číslo 3 (pretože 10, 4 a 0 sú deliteľné 2).

Volanie *pocet_delitelnych(4,0)* vypíše chybovú správu, že hodnota k nie je kladná a **vráti** -1.

Volanie *pocet_delitelnych(-1,2)* vypíše chybovú správu, že hodnota n je záporná a **vráti** -1.

Volanie *pocet_delitelnych(-1,-2)* vypíše chybovú správu, že hodnota n je záporná a že hodnota k nie je kladná a **vráti** -1.

Úloha č. 8

Definujte funkciu *sucet_nacitanych(n)*, ktorá má vstupný parameter nezáporné číslo n . Funkcia načíta n čísiel z klávesnice a **vráti** súčet načítaných čísiel. V prípade, že n je záporné číslo funkcia vypíše príslušnú chybovú správu a **vráti** hodnotu -1. V prípade, že n je nula, funkcia **vráti** hodnotu 0.

Ukážky vstupov a výstupov:

Volanie *sucet_nacitanych(-1)* vypíše chybovú správu, že hodnota n je záporná a **vráti** -1.

Volanie *sucet_nacitanych(4)* načíta 4 čísla z klávesnice. Ak by čísla boli napríklad 5, -2, 10, -7, funkcia **vráti** 6, pretože $5 + (-2) + 10 + (-7) = 6$.

Úloha č. 9

Definujte funkciu *minimum(n)*, ktorá má vstupný parameter kladné číslo n . Funkcia načíta n čísiel z klávesnice a **vráti** najmenšie načítané číslo. V prípade, že n je nula alebo záporné číslo, funkcia vypíše príslušnú chybovú správu a **vráti** hodnotu **None**.

Ukážky vstupov a výstupov:

Volanie *minimum(4)* načíta 4 čísla z klávesnice. Ak by to boli čísla 8,2,5,3 funkcia **vráti** číslo 2.

Volanie *minimum(5)* načíta 5 čísiel z klávesnice. Ak by to boli čísla 5,-2,1,2,3 funkcia **vráti** číslo -2.

Úloha č. 10

Definujte funkciu *druhe_najvacsie(n)* s parametrom celé číslo n . Funkcia načíta n čísiel z klávesnice a vráti druhé najväčšie načítané číslo (druhé maximum). V prípade, že má parameter n hodnotu menšiu ako 2, funkcia vypíše chybovú hlášku, že hodnota vstupného argumentu funkcie musí byť aspoň 2 a vráti hodnotu None.

Ukážky vstupov a výstupov:

Volanie *druhe_najvacsie(1)* vypíše chybu a vráti hodnotu None.

Volanie *druhe_najvacsie(5)* načíta 5 čísiel z klávesnice. Ak sú tieto čísla napríklad 5,-2,1,2,3 funkcia **vráti** číslo 3, pretože $-2 \leq 1 \leq 2 \leq \mathbf{3} \leq 5$

Volanie *druhe_najvacsie(3)* načíta 3 čísla z klávesnice. Ak sú tieto čísla napríklad -10, -5, 0 funkcia **vráti** číslo -5, pretože $-10 \leq \mathbf{-5} \leq 0$

Sekcia 2: Úlohy o prvočíslach

Poznámka: Úlohy v tejto sekcii na seba nadväzujú!

Úloha č. 11

Definujte funkciu *delitelnost(a,d)* so vstupnými parametrami kladnými celými číslami a a d . Funkcia **vráti** hodnotu True ak d delí a , inak vráti hodnotu False. V tejto úlohe nemusíte ošetrovať korektnosť parametrov a a d .

Hint: Číslo d delí číslo a vtedy, ak zvyšok a po delení d je rovný nule.

Ukážky vstupov a výstupov:

Volanie *delitelnost(6,3)* **vráti** hodnotu True, pretože 3 delí 6.

Volanie *delitelnost(7,3)* **vráti** hodnotu False, pretože 3 nedelí 7.

Úloha č. 12

Definujte funkciu *je_prvocislo(a)* so vstupným parametrom kladným číslom a , ktorá vráti hodnotu True, ak je parameter a prvočíslo. Ak je a zložené číslo, funkcia vráti hodnotu False. Môžete predpokladať, že a bude mať vždy hodnotu aspoň 2. V implementácii funkcie *je_prvocislo(a)* využijete funkciu *delitelnost()* z predošlej úlohy!

Hint: Celé číslo $a > 2$ je prvočíslo, ak pre všetky čísla od 2 po $a-1$ platí, že nedelia číslo a . Číslo 2 je prvočíslo automaticky.

Ukážky vstupov a výstupov:

Volanie *je_prvocislo(7)* **vráti** hodnotu True, pretože 7 je prvočíslo, pretože ho nedelí žiadne z čísiel 2,3,4,5,6.

Volanie *je_prvocislo(8)* **vráti** hodnotu False, pretože 8 nie je prvočíslo. Z čísiel 2,3,4,5,6,7 ho delia čísla 2 a 4.

Volanie *je_prvocislo(2)* **vráti** hodnotu True, pretože 2 je prvočíslo.

Sekcia 3: Šachové úlohy

Poznámka: Úlohy v tejto sekcii sú prevzaté zo stránky

https://snakify.org/en/lessons/if_then_else_conditions/problems/

POZOR! Všetky úlohy v tejto sekcii majú krátke riešenie iba na pár riadkov! Snažte sa na toto riešenie prísť!

Úloha č. 13

Definujte funkciu *pohyb_veze*($x1,y1,x2,y2$). Funkcia má 4 parametre: $x1,y1,x2,y2$, každý nadobúda hodnoty od 1 po 8. Funkcia **vráti** hodnotu True, ak sa šachová veža dokáže na 1 ťah dostať z pozície ($x1,y1$) na šachovnici do pozície ($x2,y2$). Inak funkcia **vráti** hodnotu False.

Hint: Pomôže vám obrázok na

https://snakify.org/en/lessons/if_then_else_conditions/problems/rook_move/

Ukážky vstupov a výstupov:

Volanie *pohyb_veze*(3,3,3,7) **vráti** True, volanie *pohyb_veze*(3,3,4,7) **vráti** False.

Úloha č. 14

Definujte funkciu *rovnaka_farba*($x1,y1,x2,y2$). Funkcia má 4 parametre: $x1,y1,x2,y2$, každý nadobúda hodnoty od 1 po 8. Funkcia **vráti** hodnotu True, ak má pozícia ($x1,y1$) na šachovnici rovnakú farbu ako pozícia ($x2,y2$). Inak funkcia **vráti** hodnotu False.

Hint: Pomôže vám obrázok na

https://snakify.org/en/lessons/if_then_else_conditions/problems/chess_board/

Ukážky vstupov a výstupov:

Volanie *rovnaka_farba*(4,2,3,7) **vráti** True, volanie *rovnaka_farba*(4,2,7,6) **vráti** False.

Úloha č. 15

Definujte funkciu *pohyb_krala*($x1,y1,x2,y2$). Funkcia má 4 parametre: $x1,y1,x2,y2$, každý nadobúda hodnoty od 1 po 8. Funkcia **vráti** hodnotu True, ak sa šachový kráľ dokáže na jeden ťah dostať z pozície ($x1,y1$) na šachovnici do pozície ($x2,y2$). Inak funkcia **vráti** hodnotu False.

Hint: Pomôže vám obrázok na

https://snakify.org/en/lessons/if_then_else_conditions/problems/king_move/

Ukážky vstupov a výstupov:

Volanie *pohyb_krala*(3,3,4,3) **vráti** True, volanie *pohyb_krala*(3,3,4,5) **vráti** False.

Úloha č. 16

Definujte funkciu *pohyb_strelca*($x1,y1,x2,y2$). Funkcia má 4 parametre: $x1,y1,x2,y2$, každý nadobúda hodnoty od 1 po 8. Funkcia **vráti** hodnotu True, ak sa šachový strelce dokáže na jeden ťah dostať z pozície ($x1,y1$) na šachovnici do pozície ($x2,y2$). Inak funkcia **vráti** hodnotu False.

Hint: Pomôže vám obrázok na

https://snakify.org/en/lessons/if_then_else_conditions/problems/bishop_move/

Ukážky vstupov a výstupov:

Volanie *pohyb_strelca*(3,3,1,5) **vráti** True, volanie *pohyb_strelca*(3,3,1,4) **vráti** False.

Úloha č. 17

Definujte funkciu *pohyb_damy*($x1,y1,x2,y2$). Funkcia má 4 parametre: $x1,y1,x2,y2$, každý nadobúda hodnoty od 1 po 8. Funkcia **vráti** hodnotu True, ak sa šachová dáma dokáže na jeden ťah dostať z pozície ($x1,y1$) na šachovnici do pozície ($x2,y2$). Inak funkcia **vráti** hodnotu False.

Hint: Pomôže vám obrázok na

https://snakify.org/en/lessons/if_then_else_conditions/problems/queen_move/

Ukážky vstupov a výstupov:

Volanie *pohyb_damy*(3,3,8,3) **vráti** True, volanie *pohyb_damy*(3,3,5,4) **vráti** False.

Úloha č. 18

Definujte funkciu *pohyb_jazdca*($x1,y1,x2,y2$). Funkcia má 4 parametre: $x1,y1,x2,y2$, každý nadobúda hodnoty od 1 po 8. Funkcia **vráti** hodnotu True, ak sa šachový jazdec dokáže na jeden ťah dostať z pozície ($x1,y1$) na šachovnici do pozície ($x2,y2$). Inak funkcia **vráti** hodnotu False.

Hint: Pomôže vám obrázok na

https://snakify.org/en/lessons/if_then_else_conditions/problems/knight_move/

Ukážky vstupov a výstupov:

Volanie *pohyb_jazdca*(5,4,3,3) **vráti** True, volanie *pohyb_jazdca*(5,4,6,5) **vráti** False.