

# Výsledky národných projektov IT Akadémia a DiTEdu v predmete informatika

Ľubomír Šnajder, Ján Guniš

Národný projekt Digitálna transformácia vzdelávania a školy (DiTEdu) je financovaný prostriedkami Európskej únie v rámci Programu Slovensko.

# Aktivity v predmete informatika

- **Metodické a učebné materiály**

- [Informatika pre štandardné triedy](#)

- [Programovanie v Pythone – zbierka inovatívnych metodík](#) + [príloha](#) (2021)
    - Programovanie v jazyku Python – Pracovný zošit (2022)

- [Nové informatické predmety](#)

- Informatika v prírodných vedách a matematike -> STEM projekty a skúmania

- **Vzdelávanie**

- Formálne – Inovačné vzdelávanie (50 hodín)

- Neformálne – Kluby učiteľov informatiky (prezenčné stretnutia, webináre)

# Koncepcia pracovného zošita

- Python ako hlavný jazyk, ostatné jazyky (Ai2, MakeCode)  
– iné programovanie (senzory, aktuátory, ...)
- Jazykové prostriedky vs. stratégie riešenia problémov / koncepty informatického myslenia  
(skôr funkcie a cykly, korytnačia grafika, výnimky, projekty)
- Rôzne typy metodík – bádateľská, problémová (systemizačná), tradičná, projektová
- Komplexná podpora pre učiteľa a žiaka
  - metodika, pracovné listy, pracovné súbory
  - školenia, webináre
  - ďalšia spolupráca s učiteľmi (aktualizácia, výskum)

| oblasť                                                                  | názov metodiky                                                           |
|-------------------------------------------------------------------------|--------------------------------------------------------------------------|
| vizualizácia priebehu výpočtu pomocou korytnačej grafiky (modul turtle) | 01 Úvod do programovania, výpočty v konzole                              |
|                                                                         | 02 Korytnačia grafika                                                    |
|                                                                         | 03 Vlastné funkcie bez parametrov a bez návratovej hodnoty               |
|                                                                         | 04 Cyklus s pevným počtom opakovaní                                      |
|                                                                         | 05 Opakovanie I. + didaktický test                                       |
|                                                                         | 06 Vlastné funkcie s parametrami – kresliace úlohy                       |
| konzolové aplikácie, riešenie konkrétnych problémov                     | 07 Vlastné funkcie s parametrami a výstupom – výpočtové úlohy            |
|                                                                         | 08 Chyby vo výpočtoch, ich rozpoznávanie a odstraňovanie                 |
|                                                                         | 09 Podmieneny príkaz                                                     |
|                                                                         | 10 Opakovanie II. + didaktický test                                      |
|                                                                         | 11 Reťazce                                                               |
|                                                                         | 12 Reťazcové metódy, zložené a vnorené podmienky                         |
|                                                                         | 13 Algoritmy s reťazcami                                                 |
|                                                                         | 14 Odchytávanie výnimiek                                                 |
|                                                                         | 15 Generovanie výnimiek                                                  |
|                                                                         | 16 Opakovanie III. + didaktický test                                     |
|                                                                         | 17 Zoznamy a metódy zoznamov                                             |
|                                                                         | 18 Vytváranie a modifikácia zoznamov, použitie náhody                    |
|                                                                         | 19 Algoritmy so zoznamami                                                |
|                                                                         | 20 Cyklus s podmienkou                                                   |
|                                                                         | 21 Vnorené radiace štruktúry                                             |
|                                                                         | 22 Opakovanie IV. + didaktický test                                      |
| grafické aplikácie (modul tkinter)                                      | 23 Grafické používateľské rozhranie – tlačidlá, textové polia, popisy    |
|                                                                         | 24 Grafické používateľské rozhranie – plátno                             |
|                                                                         | 25 Komplexný projekt – výber problému, analýza a návrh riešenia problému |
|                                                                         | 26 Komplexný projekt – implementácia riešenia problému                   |
|                                                                         | 27 Komplexný projekt – finalizácia projektu + prezentácia a diskusia     |

# Koncepty informatického myslenia

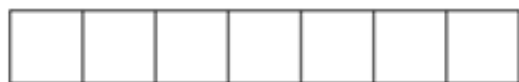
- **Logika** – predpovedaj, analyzuj
- **Algoritmy** – vytváraj kroky a pravidlá
- **Dekompozícia** – rozdeľ na časti
- **Hľadanie vzorov** – rozpoznaj a využívaj podobnosti
- **Abstrakcia** – vyber podstatné, odhliadni od menej podstatného
- **Vyhodnotenie** – rob rozhodnutia

# Ukážka bádateľskej metodiky

- **Zapojenie** (5 minút) – analýza obrázkov s opakujúcim (aj neopakujúcim) sa vzorom na pravidelných pozíciách a diskusia k spoločným črtám opisov obrázkov (úloha 1 z pracovného listu)
- **Skúmanie** (7 minút) – skúmanie hotových a upravovanie nekompletných príkazov obsahujúcich príkaz cyklu `for`, získanie prvej skúsenosti a predstave o príkaze cyklu `for` (úloha 2 z pracovného listu)
- **Vysvetlenie** (6 minút) – diskusia k ponímaniu príkazu cyklu `for` a naše zhrnutie učiva
- **Rozpracovanie** (14 minút) – programovanie obrázkov s opakujúcimi sa vzormi na pravidelne rozmiestnených pozíciách (úlohy 3 až 5 z pracovného listu)
- **Hodnotenie** (8 minút) – vyriešenie sebahodnotiaceho testu s našim vyhodnotením (úlohy 6 a 7 z pracovného listu)

**Úloha 1** Opíšte **stručne** a čo **najpresnejšie** (pre kamaráta na telefóne), čo vidíte na uvedených obrázkoch **A, B, C, D**:

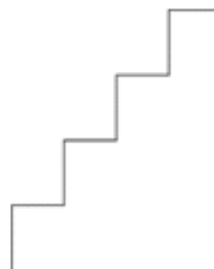
**A**



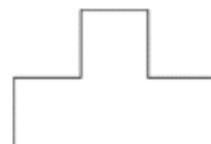
**B**



**C**



**D**



**A** .....

**B** .....

**C** .....

**D** .....

Uvedte, ktoré z obrázkov sa vám opisovali ľahšie ..... a ktoré ťažšie .....

Uvedte, čo majú spoločné vaše popisy ľahšie opísateľných obrázkov: .....

.....

**Úloha 2** Preskúmajte obidva uvedené programy **A** a **B**. Najprv len na základe prečítania uvedeného programového kódu, potom po spustení ich kódov uložených v súboroch **04\_02A.py** a **04\_02B.py**.

**A**

```
import turtle

tabula = turtle.Screen()
pero = turtle.Turtle()
pero.penup()

pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)

tabula.mainloop()
```

**B**

```
import turtle

tabula = turtle.Screen()
pero = turtle.Turtle()
pero.penup()

for i in range(5):
    pero.dot(40, 'orange')
    pero.forward(40)

tabula.mainloop()
```

- a) Opíšte obrázky, ktoré vykreslia uvedené programy **A** a **B**: .....
- .....
- b) Ktorý zo zápisov pokladáte za lepší a prečo: .....
- c) Ako by sa zmenil výsledok programu **B**, ak by sme v riadku 7 namiesto range(5) uviedli range(10): .....
- .....

# VYSVETLENIE

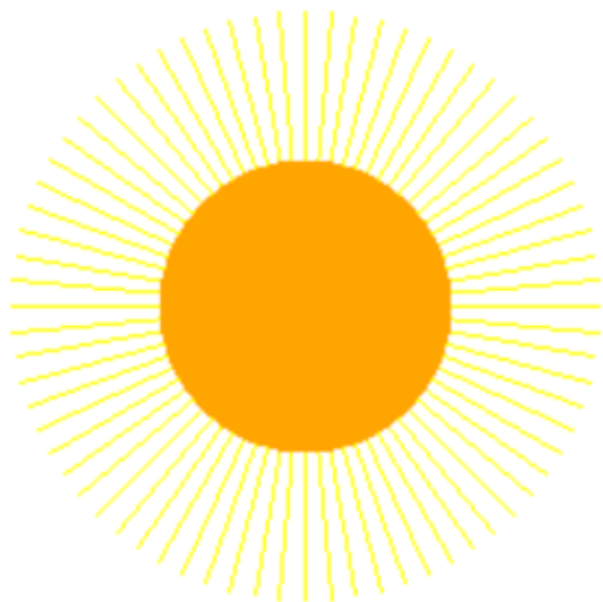
Možné otázky v diskusii:

1. Na čo je dobrý príkaz `for`?  
*(Možná odpoveď: Na skrátenie zápisu programu s opakujúcimi sa príkazmi.)*
2. Z akých častí sa skladá zápis príkazu `for`?  
*(Možná odpoveď: Riadok začínajúci slovom `for` (hlavička cyklu) a riadky s odsadenými príkazmi (telo cyklu).)*
3. Ako funguje príkaz `for`, aký význam majú jeho hlavička a telo?  
*(Možná odpoveď: V prvom riadku v parametri funkcie `range` je uvedené koľkokrát sa budú opakovať príkazy, ktoré sú odsadené v nasledovných riadkoch.)*
4. Musíme použiť v príkaze `for` vlastné funkcie?  
*(Možná odpoveď: V príkaze `for` sa môžu použiť rôzne príkazy, aj vlastné funkcie.)*

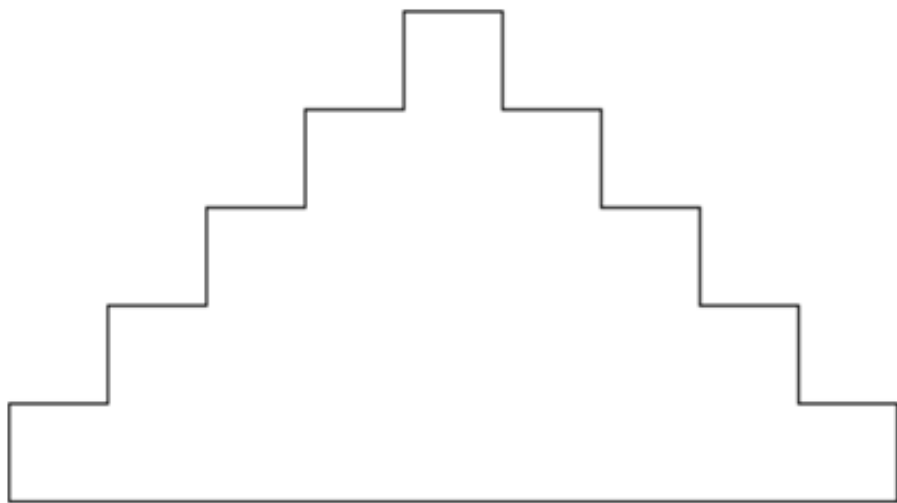
**Úloha 4** Vytvorte programy **04\_04A\_slnko.py** a **04\_04B\_schodiky.py** vykresľujúce uvedené obrázky **A** a **B** tak, aby bolo grafické pero na konci vykreslenia v počiatočnej pozícii a počiatočnom natočení.

Riešte  
podľa  
pokynov  
učiteľa

**A**



**B**



# HODNOTENIE

## Sebahodnotiaci test

**Úloha 6** Uved'te koľkokrát sa vykoná vlastná funkcia prikaz() v programoch **A**, **B** a **C**.

**A**

```
for i in range(3):  
    prikaz()  
    prikaz()  
prikaz()
```

**B**

```
for i in range(3):  
    prikaz()  
  
prikaz()
```

**C**

```
for i in range(3):  
    prikaz()  
prikaz()  
prikaz()
```

## VEDOMOSTI V KOCKE

Príkaz `for` sa využíva pri riešení problémov, ktoré pozostávajú z viacerých rovnakých (alebo podobných) podproblémov. Dá sa prirovnať k maliarskemu valčeku, ktorý pri otáčaní opakovane odtláča farbu so zadaným vzorom na stenu. Príkaz `for` sa tiež označuje ako **príkaz s pevným (so známym) počtom opakovaní** alebo aj skrátené **cyklus for**.

| Poradie                | Zápis kódu bez príkazu cyklu FOR                 | Zápis kódu s príkazom cyklu FOR                  |
|------------------------|--------------------------------------------------|--------------------------------------------------|
| 1<br>2<br>3<br>4       | príkazy()<br>príkazy()<br>príkazy()<br>príkazy() | <code>for i in range(4):</code><br>príkazy()     |
| 1<br>2<br>...<br>počet | príkazy()<br>príkazy()<br>...<br>príkazy()       | <code>for i in range(počet):</code><br>príkazy() |

# Spätná väzba od učiteľov (1)

- „Odporúčam pripravenú metodiku realizovať na **dvoch vyučovacích hodinách** a prejsť všetky úlohy aj nepovinné (8-12), aby žiaci mali možnosť **precvičiť si učivo**.“
- „V pracovnom liste navrhujem **vymeniť poradie úloh** č. 3 a č. 4, aby žiaci najprv použili cyklus úpravou programov a až potom vytvárali program vlastný“
- „Navrhujem rovno vykresľovať tieto obrázky v pythone **Tkinter**, nie v turtle, je to pre mňa návrat k imaginu :D“
- „Úlohy sú dostatočne zamerané na pochopenie cyklov. Možno by som zdôraznil aj **cyklus v cykle**.“

## Spätná väzba od učiteľov (2)

- „Žiaci majú **väčší problém s návrhom riešenia** na papier, s uvedením počiatočného a koncového bodu vykresľovania a natočenia pera, **s naprogramovaním úlohy už problém nemajú.**“
- „Musím kresliť, pretože niektorým žiakom **chýba predstavivosť**“
- „Úlohy na bádanie a objavovanie cyklu sú pekné, v prvej úlohe mali žiaci **problém opísať obrázky**. Väčšina vychádzala z bežného života a **opisovala** analógiu obrázku zo života, **nie algoritmicky**. Úloha 5 nie je až taká triviálna, ako sa na prvý pohľad zdá, a možno na cyklus by bolo vhodné vymyslieť **menej náročnú (aj časovo) úlohu.**“

# DiTEdu – tím informatika

- **Šnajder, Guniš, Andrejková, Tkáčová, Sokol, Regeciová, Krajči, Gurský, Antoni (UPJŠ)**
- **Kalaš, Tomcsányiová, Hrušecká, Hrušecký, Míková (UK)**
- **Lovászová, Michaličková (UKF)**
- **Voštinár, Melicherčík (UMB)**
- **Varga, Adamko (UNIZA)**
- **Biñas (TUCE)**
- **Beták, Očenáš (NIVAM)**

# Metodické a učebné materiály – informatika

- Nové metodiky
  - **Analýza dát** (4 h – Guniš, UPJŠ)
  - **Informatika a hudba** (3 h – Šnajder, UPJŠ)
  - **Programovanie 3D objektov** (3 h – Voštinár, UMB)
  - **Jazyk Scratch a Princípy robotiky** (9 h – Očenáš, NIVAM)
  - **3D vizualizácia a 3D tlač** (18 h – Beták, NIVAM)
- Aktualizované metodiky
  - **Informačná bezpečnosť** (15 h – Sokol, Regeciová, UPJŠ)
  - **Umelá inteligencia** (6 h – Tkáčová, UPJŠ)
  - **Rastrová grafika** (4 h – Voštinár, UMB)
  - súbor metodík pre **predmet STEMPaS** (15 h)

# Séria metodík Informatika a hudba

- Gymnázium so zameraním na informatiku, 3. ročník, 3 VH
- Podľa vzdelávacieho programu 6 oblastí informatiky:
  - Algoritmy a programovanie,
  - Práca s dátami a ich reprezentácie,
  - Systémy pre komunikáciu a interakciu,
  - Počítačové systémy a siete,
  - Informatika a spoločnosť,
  - Aplikácie informatiky v rôznych oblastiach poznania.
- produktívnym spôsobom prepája informatiku s ďalšími predmetmi alebo oblasťami poznania,

# Aplikácie informatiky v rôznych oblastiach poznania

- Prepojenie informatiky s ďalšou oblasťou má ilustrovať užitočnosť informatiky a programovania pre bádanie v iných oblastiach. Vzdelávací obsah spadajúci do tohto tematického okruhu rozširuje poznanie žiakov aj v samotnej informatike, aj v bádatej oblasti.
- Pri naplňaní vzdelávacích cieľov tohto okruhu je žiadúce:
  - organizovať výučbu v moduloch, napr. ako jeden alebo viacero modulov v celkovom rozsahu 30 vyučovacích hodín,
  - organizovať výučbu projektovou formou, najmä tímovým projektom alebo projektmi v spolupráci s konzultantmi zo skúmaných oblastí,
  - využiť príležitosť na ilustrovanie zaujímavých, moderných tém vývoja a výskumu napr. v kontexte STE(A)M vzdelávania,
  - aby sa vzdelávací obsah nerealizoval formou výkladu, ale mal formu tvorivej bádateľskej činnosti s etapou reflexie. Žiaci pri práci spravidla používajú programovanie ako nástroj na svoje bádanie.

# Vzdelávacie ciele metodík Informatika a hudba

- vysvetliť vzájomné súvislosti medzi odpovedajúcimi konceptami programovania (postupnosť príkazov, paralelné procesy, cykly, vetvenie, podprogramy) a hudby (notový zápis jednotlivých tónov, akordy, opakovanie, voltové zátvorky, refrén),
- vysvetliť spôsob digitálneho kódovania skladby (tempo, dynamika) a jej jednotlivých tónov (výška, dĺžka, farba, hlasitosť),
- prepísať skladbu z partitúry do MIDI kódovania pre jej prehratie pomocou programu,
- využívať príkazy modulu scamp a ďalších relevantných modulov pri programovaní hudby,
- programovať hudobné projekty zamerané na prehrávanie a vytváranie notového zápisu známych skladieb,
- programovať hudobné projekty zamerané na generovanie vlastných skladieb,
- programovať hudobné projekty ako pomocné nástroje pri zlepšovaní interpretačných speváckych schopností

# Použité digitálne technológie a zdroje údajov

- interpreter jazyka Python (<https://www.python.org/downloads/>),
- vývojové prostredie PyCharm Community Edition (<https://www.jetbrains.com/pycharm/download/>),
- alternatívne vývojové prostredie Thonny (<https://thonny.org/>),
- textový editor a editor zdrojových kódov PSPad (<https://www.pspad.com/>),
- veľký zdroj notových zápisov hudobných skladieb od M. Jurčiho (<https://pesnicky.orava.sk/>),

# Obsah série metodík Informatika a hudba

- **Kódovanie hudby a jej programovanie (BOV)**

- Prepojenie konceptov programovania a hudby. MIDI kódovanie skladieb. Tvorba programov prehrávajúcich skladby uložené v zoznamoch a súboroch.

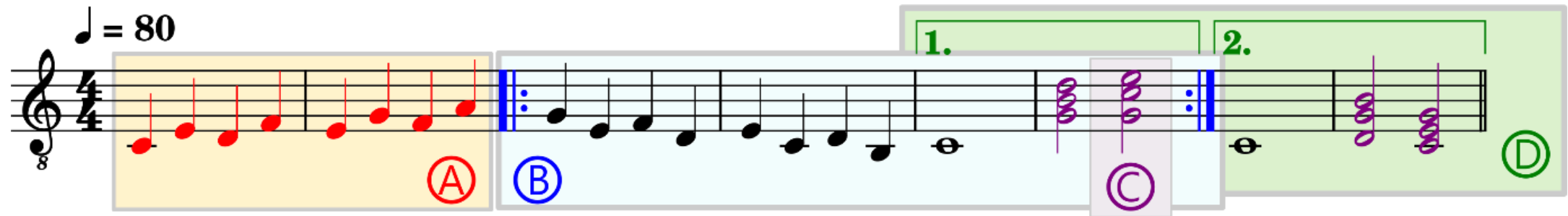
- **Prehrávanie viachlasnej skladby a tlačenie skladieb (BOV)**

- Kódovanie akordov. Tvorba programov prehrávajúcich viachlasné skladby. Tvorba programov na tlačenie skladieb.

- **Hudobná kompozičná zvedavosť (PV)**

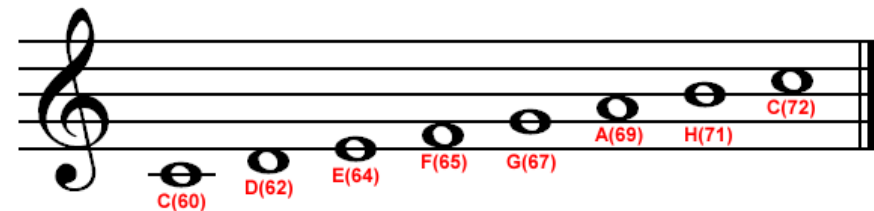
- Generovanie skladieb s použitím reťazcov, zoznamov, rekurzie, náhody. Zisťovanie intonačnej náročnosti skladieb.

# Koncepty hudby a koncepty programovania



- Ⓐ postupnosti nôt v programovaní odpovedá **postupnosť príkazov**,
- Ⓑ opakovaniu taktov s notami v programovaní odpovedá **príkaz opakovania**,
- Ⓒ alternatívnemu prehrávaniu nôt pri prvom a druhom opakovaní (prima volta, sekunda volta) v programovaní odpovedá **príkaz vetvenia**,
- Ⓓ súbežnému prehratie troch a viacerých nôt (akordu) v programovaní odpovedajú **paralelné procesy**.

# MIDI kódovanie nôt v partitúre



$\text{♩} = 110$

Pr - ší, pr - ší, len sa le - je, ne - za - tvá - raj mi - lá dve - re,

mi - lá má, du - ša má, ne - za - tvá - raj pred na - ma.



- $4^*[67, 67, 69, 67], 2^*[3^*[65], 3^*[64], 62, 62, 67, 67, 3^*[60], \text{pauza}]$
- $[[67, 0.5], [67, 0.5], [69, 0.5], [67, 0.5], [67, 0.5], [67, 0.5], [69, 0.5], [67, 0.5], [67, 0.5], [67, 0.5], [69, 0.5], [67, 0.5], [67, 0.5], [67, 0.5], [69, 0.5], [67, 0.5], [65, 0.5], [65, 0.5], [65, 1], [64, 0.5], [64, 0.5], [64, 1], [62, 0.5], [62, 0.5], [67, 0.5], [67, 0.5], [60, 0.5], [60, 0.5], [60, 0.5], [\text{pauza}, 0.5], [65, 0.5], [65, 0.5], [65, 1], [64, 0.5], [64, 0.5], [64, 1], [62, 0.5], [62, 0.5], [67, 0.5], [67, 0.5], [60, 0.5], [60, 0.5], [60, 0.5], [\text{pauza}, 0.5]]$

# Prehratie, noty, pauzy, akordu

```
1 from scamp import *
2
3 s = Session()
4 piano = s.new_part('piano')
5 piano.play_note(pitch: 60, volume: 1, length: 0.5)
6 wait(2)
7 for vyska in [60, 64, 67]:
8     piano.play_note(vyska, volume: 0.5, length: 0.5)
9 wait(2)
10 piano.play_chord(pitches: [60, 64, 67], volume: 0.8, length: 2.0)
```

... ukážky programov (paralelné prehrávanie skladieb, generovanie skladieb, tvorba pomôcok pre spev, výpočet náročnosti skladby)

# Kluby učiteľov informatiky – prezenčné stretnutia

- **Kvantové počítače a kvantové programovanie** (24.09.24, Doboš, Buchta)
- **Robotické stavebnice na 1. stupni ZŠ** (25.10.24, Voštinár, Melicherčík)
- **STEAM vyučovanie informatiky** (19.11.24, Šnajder, Guniš)
- **Ako riešiť kybernetické bezpečnostné incidenty** (10.12.24, Sokol, Regeciová)
- **Rozšírená realita v prostredí CoSpaces** (18.12.24, Voštinár, Melicherčík)
- **Responzívny web** (16.01.25, Gurský)

# KUI – webináre k novým a aktualiz. metodikám

- **Vizualizácia pomocou Lego súčiastok** (13.09.24, Beták)
- **Vybrané slová s jazykom SCRATCH** (18.09.24, Očenášš)
- **Rozprávky s ozobotmi – Drak a princezná** (19.09.24, Očenášš)
- **Dotyková mapa Slovenska** (07.11.24, Očenášš)
- **Úvod do 3D modelovania a 3D tlače** (21.11.24, 20,01,25 Beták)
- **Bludiská s ozobotmi** (23.0.,25, Očenášš)
- ...

# KUI – webináre k novým a aktualiz. metodikám

- **Rastrová grafika** (04.02.25, Voštinár)
- **Načo sú nám dáta – personalizácia obsahu, ELSA** (10.04.25, Tokárová, Tkáčová)
- **Objavujeme umelú inteligenciu – negeneratívna, generatívna** (03.04.25, Prievozník)
- **Ako sa učí počítač – základy, aplikácie** (02.04.25, Štefanková)
- **Informatika a hudba 1-3** (11.04.25, Šnajder)
- **Analýza dát 1-4** (08.04.25, Guniš)

# KUI – webináre k aktuálnym témam informatiky

- **Raspberry Pi Pico W 101** (02.10.24, Biňas)
- **AI pre učiteľov SOŠ I. (Tajomstvá AI)** (17.10.24, Štefanková)
- **Octostudio** (24.10.24, Tkáčová)
- **Úvodné školenie do AI pre ZŠ (Tajomstvá AI)** (12.11.24, Štefanková)
- **Experimenty so zvukom, hudbou, kódom a AI** (13.11.24, Tkáčová)
- **AI pre učiteľov SOŠ II. (Tajomstvá AI)** (19.11.24, Štefanková)
- **Stavové stroje** (20.11.24, Biňas)
- **Dataseťy a práca s nimi** (05.12.24, Tkáčová)

# KUI – webináre k aktuálnym témam informatiky

- **Dátové štruktúry v Python-e** (09.12.24, Guniš)
- **Aktivity kariérovej výchovy na hodinách informatiky** (16.01.25, Tkáčová)
- **Programovanie mobilných aplikácií so zameraním na využitie senzorov a multimédií mobilného zariadenia** (12.12.24, Šnajder)
- **Chyby a manažment výnimiek v jazyku Python** (27.01.25, Guniš)
- **Programovanie mobilných aplikácií so zameraním na prácu s dátami** (29.01.25, Šnajder)
- **Špecifiká aritmetiky počítača** (26.02.25, Šnajder)
- **Súbory v Python-e a otvorené dáta** (13.05.25, Guniš)

# KUI – webináre k aktuálnym témam informatiky

- **Robotická súťaž WRO** (03.02.25, Voštinár, Melicherčík)
- **Novinky v tvorbe a dizajne webu a digitálnych aplikácií** (11.02.25, Tkáčová)
- **Creative coding** (11.03.25, Tkáčová)
- **Objektovo orientované programovanie v Pythone** (25.03.25, Guniš)
- **Programovanie matematických modelov prírodných a spoločenských javov v Pythone** (07.05.25, Šnajder)
- **Programovanie v Asymptote** (11.06.25, Krajči)
- **Databázy** (18.06.25, Krajči, Gurský)
- **Informatika na 1. stupni: Radosť z dobrého štartu** (16.06.25, Kalaš)



Digitálna transformácia vzdelávania a školy



Národné centrum  
pre digitálnu transformáciu vzdelávania



NÁRODNÝ INŠTITÚT VZDELÁVANIA A MLÁDEŽE



UNIVERZITA  
PAVLA JOZEFA ŠAFÁRIKA  
V KOŠICIACH



UNIVERZITA  
KOMENSKÉHO  
V BRATISLAVE



MINISTERSTVO  
ŠKOLSTVA, VÝSKUMU,  
VÝVOJA A MLÁDEŽE  
SLOVENSKEJ REPUBLIKY

# Ďakujeme za pozornosť

Kontakt:

- Ľubomír Šnajder, [lubomir.snajder\(at\)upjs.sk](mailto:lubomir.snajder@upjs.sk)
- Ján Guniš, [jan.gunis\(at\)upjs.sk](mailto:jan.gunis@upjs.sk)