

# Interaktívny pracovný zošit pre výučbu logiky pre informatikov

Matej Mok  
Školiteľ: Mgr. Ján Klúka, PhD.

# Úvod

- Vývoj aplikácie pre podporu výuky matematickej logiky
- Teoretické a praktické úlohy
- Praktické úlohy odovzdávané pomocou Githubu
- Zjednodušenie práce na teoretických úlohách

Basic propositional • Prettyfy formulas • Print • Export as JSON • Import from JSON • Undo • Redo

- $\neg ( \text{pocuva}(\text{Fero}, \text{PinkFloyd}) \vee \text{hra}(\text{Fero}, \text{FerovaPS}) )$   $S^*$  [ ] ...
- $(\text{PinkFloyd}) \wedge \text{hraciaKonzola}(\text{FerovaPS})$   $S^*$  [ ] ...
- $\text{rustrovany}(\text{Fero}) \rightarrow \text{kino}(\text{Fero}, \text{Anka})$   $S^*$  [ ] ...
- $\rightarrow ( \text{pocuva}(\text{Fero}, \text{PinkFloyd}) \vee \text{hra}(\text{Fero}, \text{FerovaPS}) )$   $S^*$  [ ] ...
- $\wedge ( \text{pocuva}(\text{Fero}, \text{PinkFloyd}) \wedge \text{hra}(\text{Fero}, \text{FerovaPS}) )$   $S^*$  [ ] ...
- $\text{ro}, \text{FerovaPS}) \rightarrow \text{pocuva}(\text{Fero}, \text{PinkFloyd})$   $S^*$  [ ] ...
- $\text{a}(\text{Fero}, \text{PinkFloyd}) \rightarrow \neg \text{frustrovany}(\text{Fero})$   $S^*$  [ ] ...
- $\text{a}(\text{Fero}, \text{FerovaPS}) \rightarrow \text{kino}(\text{Fero}, \text{Anka})$   $S^*$  [ ] ...

{ } Množiny Graf Exportovať Importovať

Jazyk  $\mathcal{L}$  Hotovo ?

Individuové konštanty  
 $C_c = \{ \text{Curo}, \text{Filip}, \text{Felix}, \text{Granule}, \text{Kosti}, \text{Kuracie\_prsia} \}$

Predikátové symboly  
 $P_c = \{ \text{kocur}/1, \text{ma\_rad}/2, \text{zozral}/2 \}$

Funkčné symboly  
 $F_c = \{ \text{lce}/1 \}$

Pravdivosť f

- $\varphi_1$  lexists y k  $\mathcal{M} \models$
- $\varphi_2$  lexists j (  $\mathcal{M} \models$

+ Pridať

Undo Redo

Language

Constants Dunčo

Functions

Predicates šteká/1, pes/1, hryzie/1

Proof

- 1  $\text{šteká}(x) \vee \neg \text{pes}(x)$
- 2  $\neg \text{pes}(x) \vee \text{hryzie}(x)$

# Požiadavky

- Editor formátovaného textu + matematika
- Zjednodušenie používania existujúcich interaktívnych aplikácií
- Vytváranie zadaní
- Prehliadanie zadaní
- Integrácia s Githubom

# Inšpirácia existujúcimi riešeniami

- Jupyter Notebook, Colaboratory

Welcome To Colaboratory

Súbor Upraviť Zobraziť Vložiť Runtime Nástroje Pomocník

+ Kód + Text Kopírovať na Disk

Pripojiť Úpravy

- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) to learn more, or just get started below!

## Getting started

The document you are reading is not a static web page, but an interactive environment called a **Colab notebook** that lets you write and execute code.

For example, here is a **code cell** with a short Python script that computes a value, stores it in a variable, and prints the result:

```
[ ] seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day
```

86400

To execute the code in the above cell, select it with a click and then either press the play button to the left of the code, or use the keyboard shortcut "Command/Ctrl+Enter". To edit the code, just click the cell and start editing.

Variables that you define in one cell can later be used in other cells:

Jupyter Beyond Plain Python (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted Python 3

Memory: 252 MB / 2 GB

## IPython: beyond plain Python

When executing code in IPython, all valid Python syntax works as-is, but IPython provides a number of features designed to make the interactive experience more fluid and efficient.

### First things first: running code, getting help

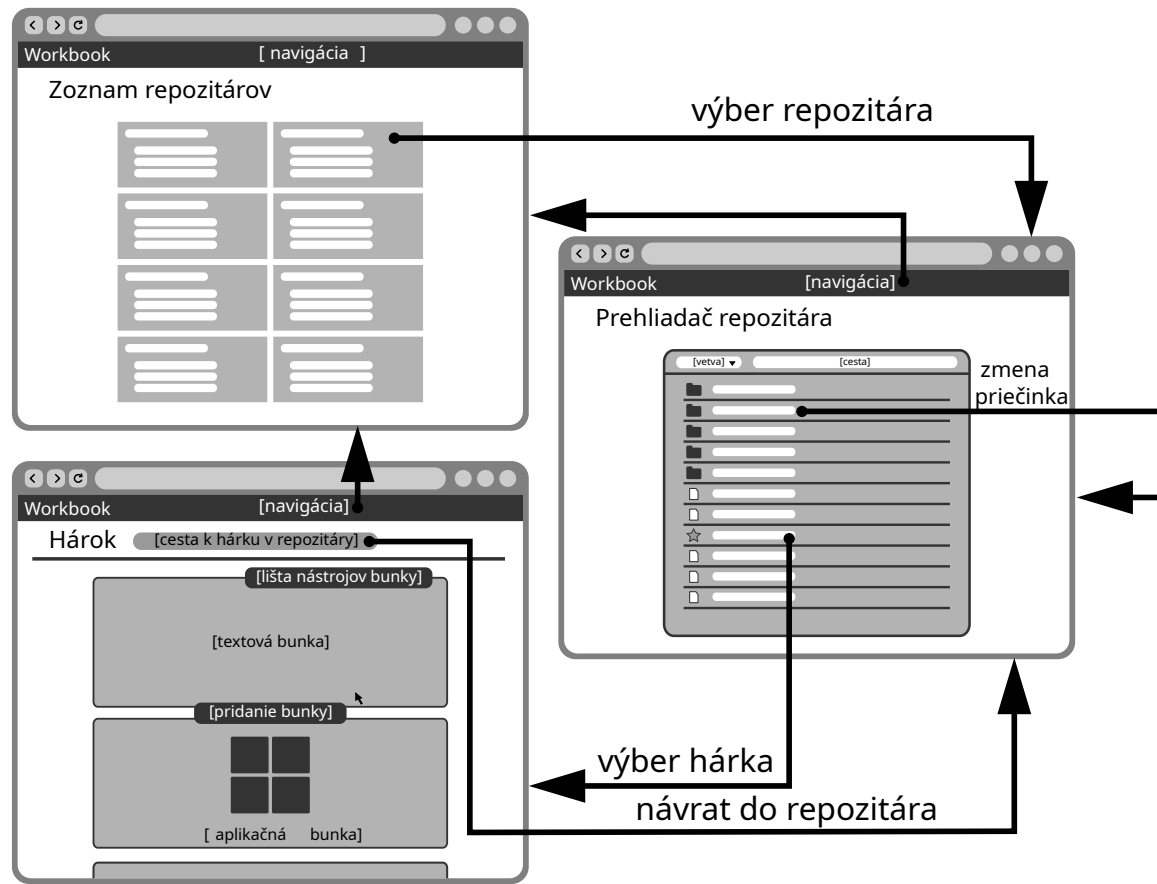
In the notebook, to run a cell of code, hit `Shift-Enter`. This executes the cell and puts the cursor in the next cell below, or makes a new one if you are at the end. Alternately, you can use:

- `Alt-Enter` to force the creation of a new cell unconditionally (useful when inserting new content in the middle of an existing notebook).
- `Control-Enter` executes the cell and keeps the cursor in the same cell, useful for quick experimentation of snippets that you don't need to keep permanently.

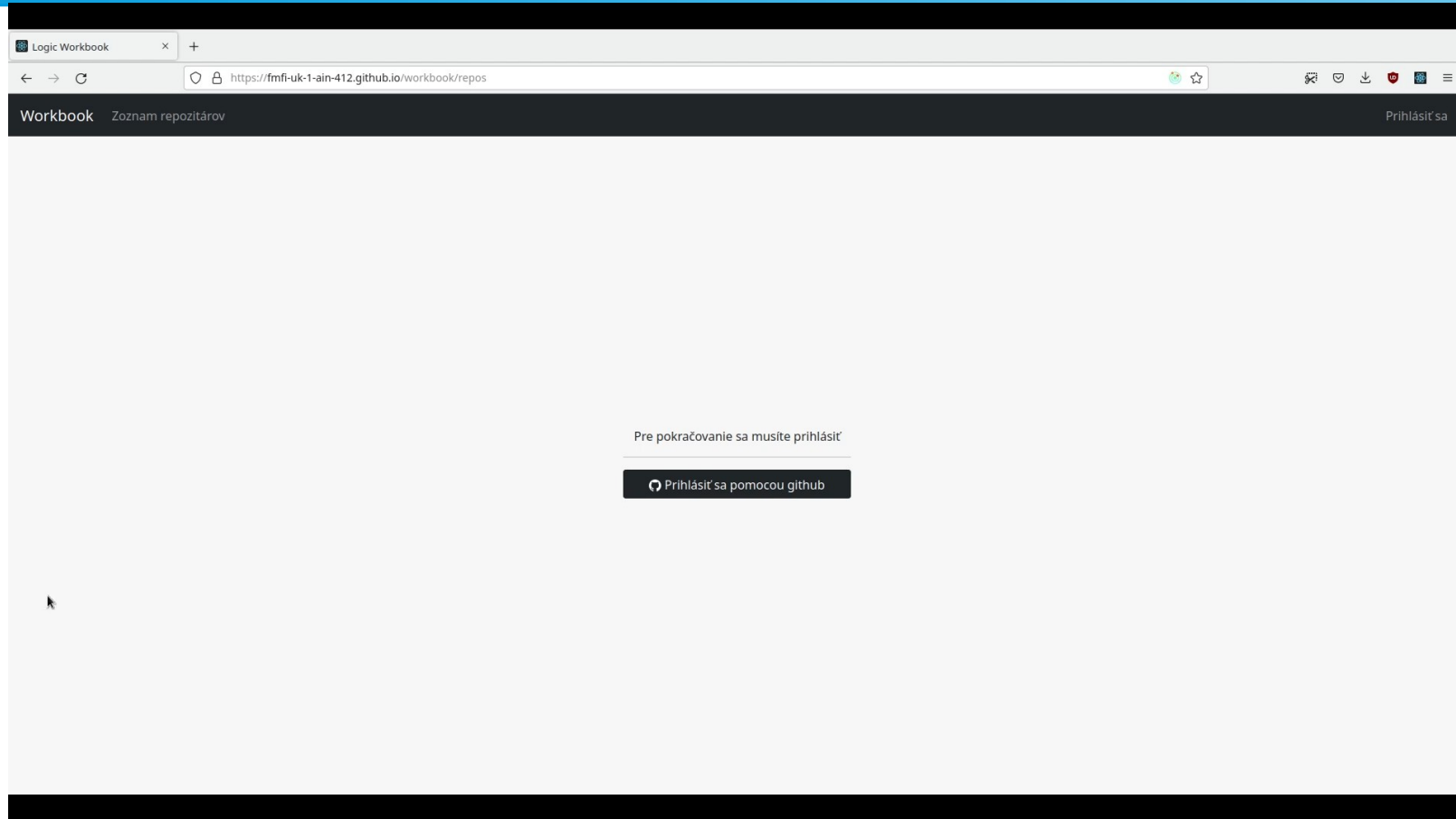
```
In [1]: print("Hi")
Hi
```

# Návrh používateľského rozhrania

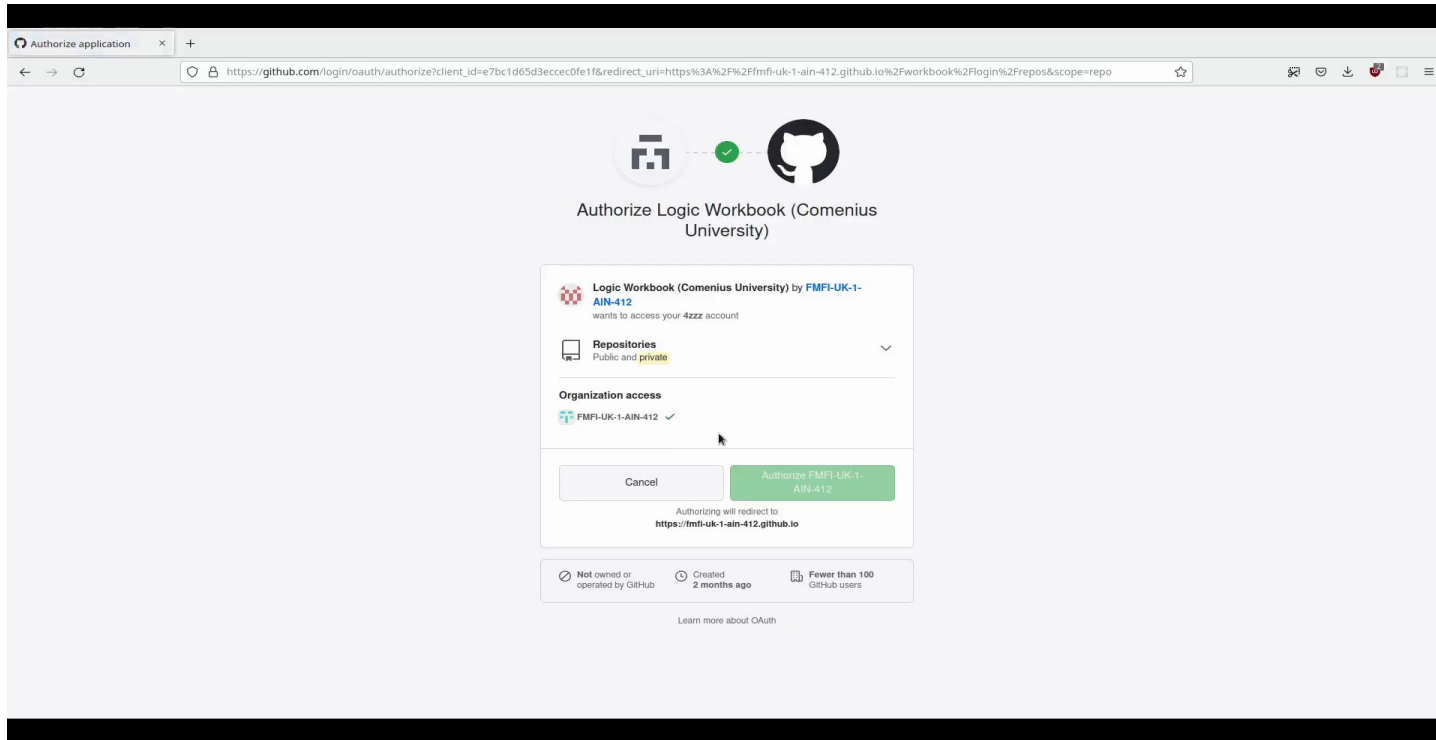
- Vytváranie zadaní a ich zobrazenie
  - Zoznam repozitárov
  - Prehliadač repozitára
- Písanie zadaní a riešení
  - Editor pracovného hárku



# Demonštrácia



# Demonštrácia

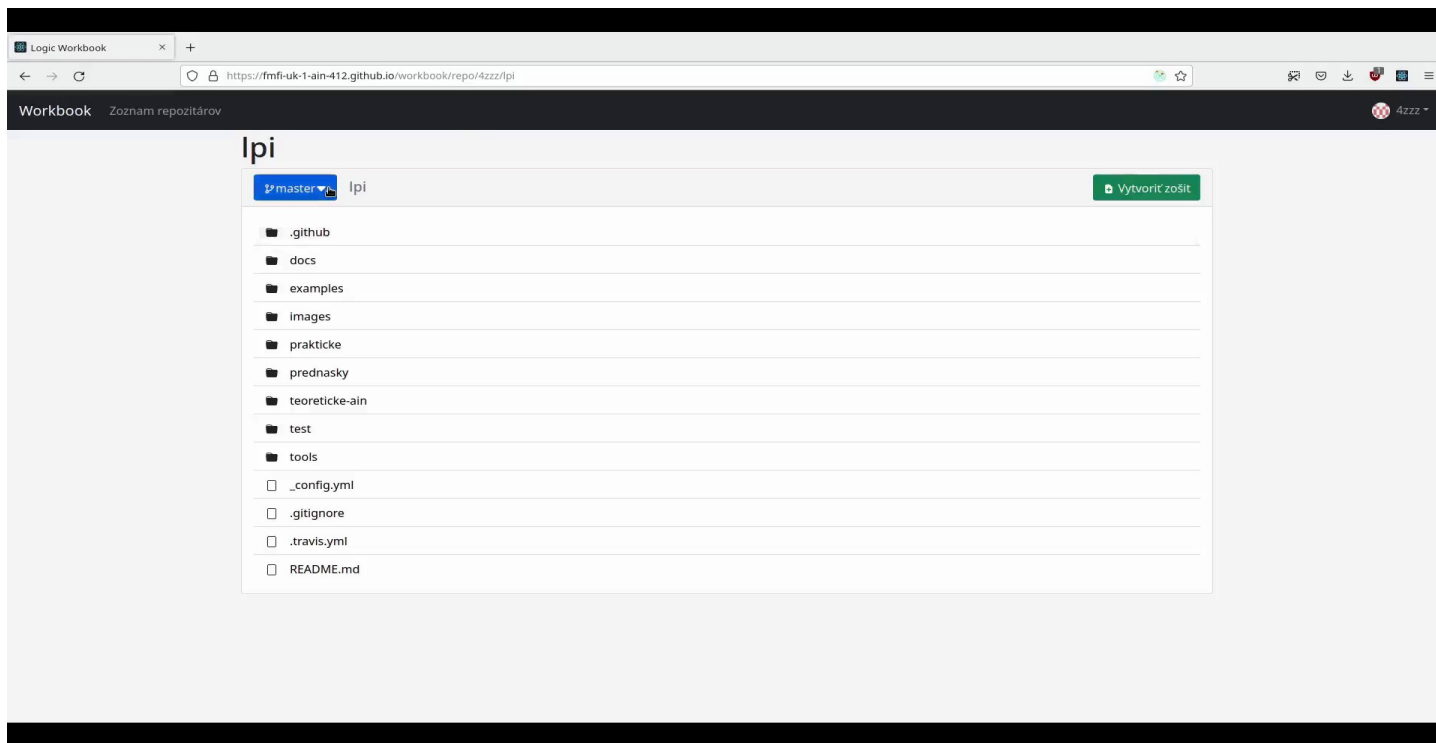


# Demonštrácia

The screenshot shows a web browser window with the address bar displaying `https://fmfi-uk-1-ain-412.github.io/workbook/repos`. The page title is "Workbook" and the subtitle is "Zoznam repozitárov". The main content area is titled "Zoznam repozitárov" and features a search bar labeled "Hľadať repozitáre" and two buttons: "Zoradenie" and "Repozitáre". Below the search bar, there is a grid of repository cards. Each card displays the repository name, a brief description, a status label (e.g., "Verejný" or "Súkromný"), and the time since it was last updated.

Repository Name	Description	Status	Last Updated
<a href="#">lpi</a>	Logika pre informatikov	Verejný	Zmenené pred pár sekundami
<a href="#">workbook-exercise</a>	Bez popisu	Verejný	Zmenené pred 6 hodinami
<a href="#">webpack-react-component-library</a>	Bez popisu	Verejný	Zmenené pred 7 hodinami
<a href="#">isolate-css</a>	Bez popisu	Verejný	Zmenené pred 12 dňami
<a href="#">bc-text</a>	Bez popisu	Súkromný	Zmenené pred 12 dňami
<a href="#">structure-explorer</a>	Structure explorer for first order logic	Verejný	Zmenené pred 12 dňami
<a href="#">resolution-editor</a>	An editor of first-order resolution proofs (refutations)	Verejný	Zmenené pred 12 dňami
<a href="#">rtk-query-github-api</a>	Bez popisu	Verejný	Zmenené pred 17 dňami
<a href="#">tableauEditor</a>	First-order tableau editor	Verejný	Zmenené pred mesiacom
<a href="#">workbook-auth-backend</a>	Bez popisu	Verejný	Zmenené pred mesiacom

# Demonštrácia

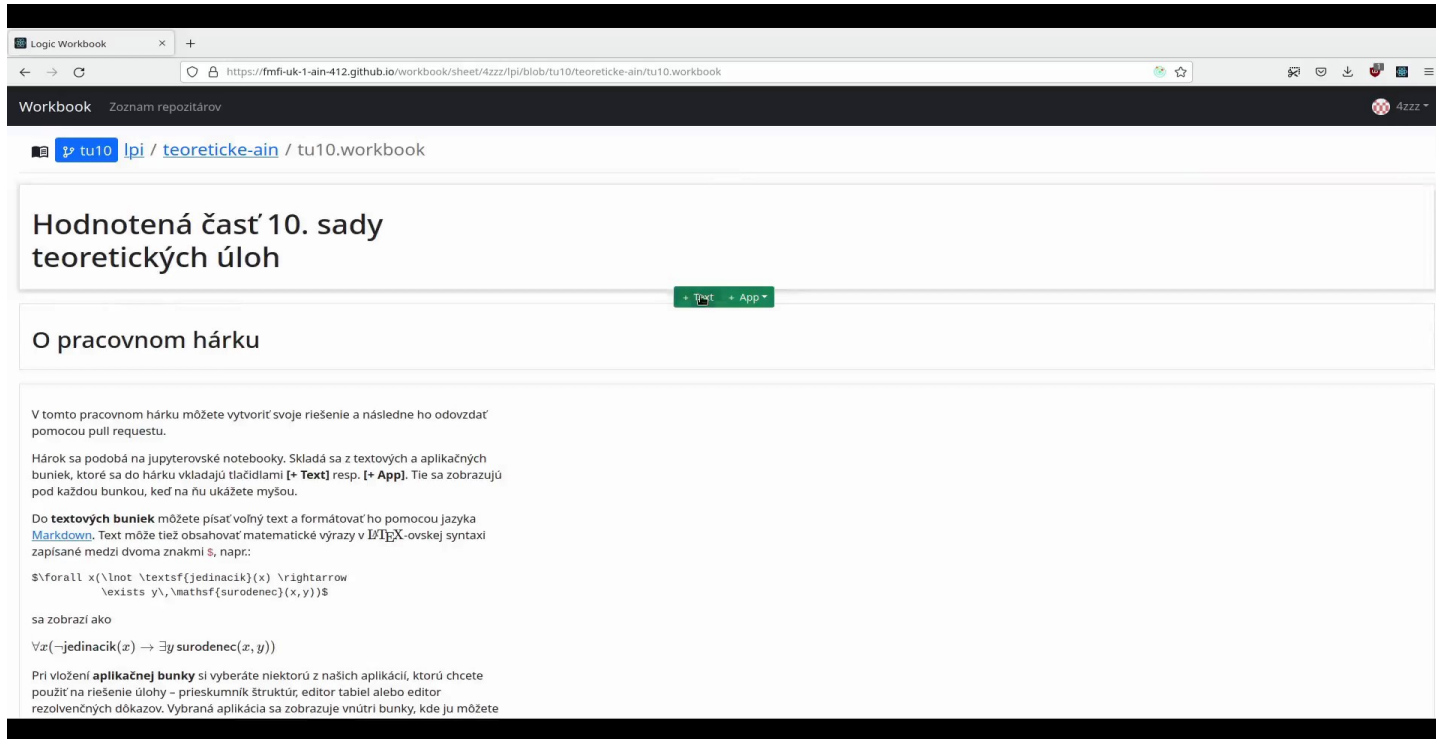


# Demonštrácia

The screenshot shows a web browser window with the following elements:

- Browser Tab:** Logic Workbook
- Address Bar:** <https://fmfi-uk-1-ain-412.github.io/workbook/repo/4zzz/lpi/tree/tu10/teoreticke-ain>
- Page Header:** Workbook Zoznam repozitárov
- Page Title:** lpi
- Navigation:** A dropdown menu showing 'tu10' and a breadcrumb 'lpi / teoreticke-ain'. A green button 'Vytvorit zošit' is visible in the top right.
- File List:**
  - ..
  - README.md
  - tu01.pdf
  - tu02.pdf
  - tu03.pdf
  - tu04.pdf
  - tu05.pdf
  - tu06.pdf
  - tu07.pdf
  - tu08.pdf
  - tu09.pdf
  - tu10.pdf
  - ★ [tu10.workbook](#) (highlighted)
  - tu11.pdf
  - zbierka.pdf
- Footer:** <https://fmfi-uk-1-ain-412.github.io/workbook/sheet/4zzz/lpi/blob/tu10/teoreticke-ain/tu10.workbook>

# Demonštrácia



The screenshot shows a web browser window with the address bar displaying `https://fmfi-uk-1-ain-412.github.io/workbook/sheet/4zzz/lpi/blob/tu10/teoreticke-ain/tu10.workbook`. The page title is "Workbook" and the breadcrumb is "tu10 / lpi / theoreticke-ain / tu10.workbook".

## Hodnotená časť 10. sady teoretických úloh

O pracovnom hárku

V tomto pracovnom hárku môžete vytvoriť svoje riešenie a následne ho odovzdať pomocou pull requestu.

Hárk sa podobá na jupyterovské notebooky. Skladá sa z textových a aplikačných buniek, ktoré sa do hárku vkladajú tlačidlami [+ Text] resp. [+ App]. Tie sa zobrazujú pod každou bunkou, keď na ňu ukážete myšou.

Do **textových buniek** môžete písať voľný text a formátovať ho pomocou jazyka [Markdown](#). Text môže tiež obsahovať matematické výrazy v  $\text{L}^{\text{T}}\text{E}^{\text{X}}$ -ovskej syntaxi zapísané medzi dvoma znakmi `$`, napr.:

$$\forall x (\neg \text{Jednacik}(x) \rightarrow \exists y \text{Surodenec}(x, y))$$

sa zobrazí ako

$$\forall x (\neg \text{jednacik}(x) \rightarrow \exists y \text{surodenec}(x, y))$$

Pri vložení **aplikačnej bunky** si vyberáte niektorú z našich aplikácií, ktorú chcete použiť na riešenie úlohy – prieskumník štruktúr, editor tabiel alebo editor rezolvenčných dôkazov. Vybraná aplikácia sa zobrazuje vnútri bunky, kde ju môžete

# Demonštrácia

Logic Workbook

← → ↻ <https://fmi-uk-1-ain-412.github.io/workbook/sheet/4zzz/pi/blob/tu10/teoreticke-ain/tu10.workbook>

7. Pokiaľ ide o knihomoľovu najobľúbenejšiu knihu, tak ju chce, aj ak už má knihu s rovnakým názvom. Inak dve knihy s rovnakým názvom nechce.

**Návod.** Riešenie zapíšte do novej textovej bunky za touto bunkou. ↓

1.  $\forall x \forall y : knihomol(x) \leftrightarrow (ma\_knihu(y) \rightarrow precital(x, y))$

Upraviť 🗑️ ⏪ ⏩ 🖥️

{ } Množiny Graf Exportovať Importovať

**Jazyk  $\mathcal{L}$**  ✓ Hotovo ?

Individuové konštanty  
 $C_c = \{ c1, c2 \}$

Predikátové symboly  
 $P_c = \{ \}$

Funkčné symboly  
 $F_c = \{ \}$

**Štruktúra  $\mathcal{M} = (D, I)$**  ?

Doména  
 $D = \{ \}$   
Doména nesmie byť prázdna

Interpretácia individuových konštánt  
 $I(c1) =$  Vyber hodnotu ...

Pravdivosť formúl v štruktúre  $\mathcal{M}$  ?  
+ Pridať

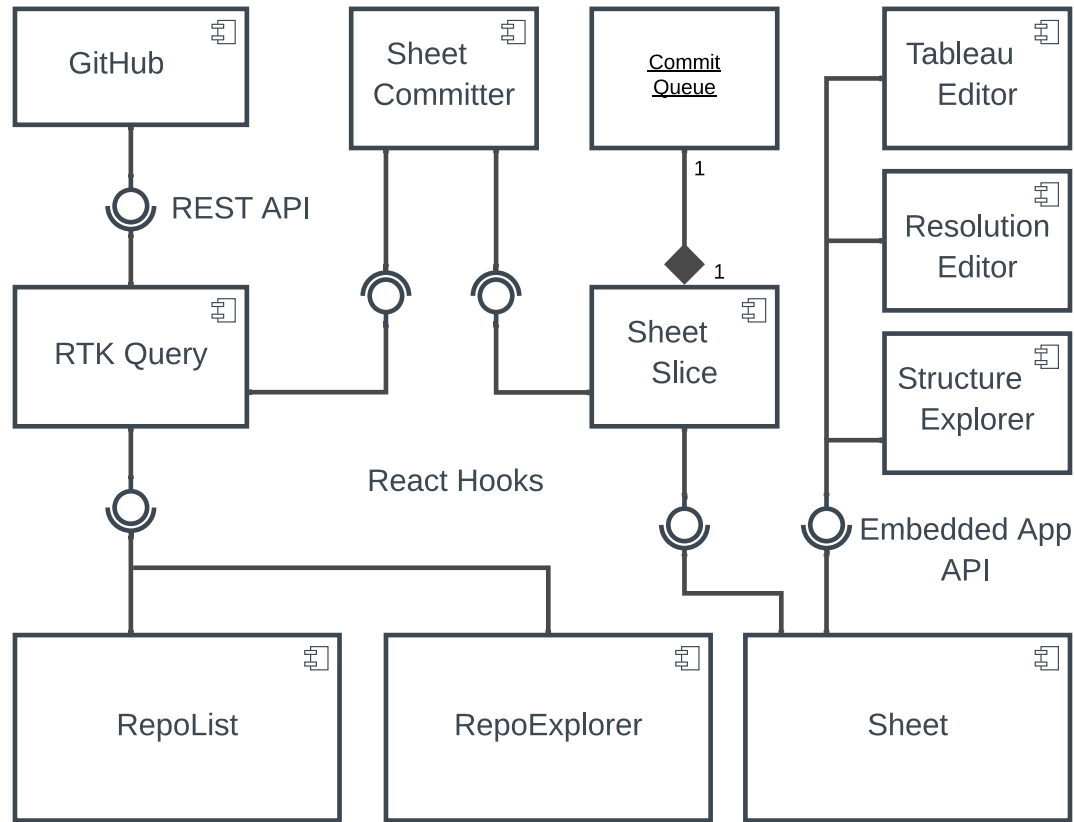
Hodnoty termov v  $\mathcal{M}$  ?  
+ Pridať

4zzz pred pár sekundami 🗑️  
OK 10/10

# Použité technológie

- Markdown
- Latex
- Git, Github
- React
- Redux
- Bootstrap 5

# Návrh a implementácia



# Rezolvenčný editor

- JavaScript
- React, Redux
- Bootstrap 4

## Resolution Editor

~ Undo Redo ~ Import Export

Language

Constants

Functions

Predicates

Proof

1	$\text{šteká}(x) \vee \neg \text{pes}(x)$	✓	Assumption	+	↑	↓	X
2	$\neg \text{pes}(x) \vee \text{hryzie}(x)$	✓	Assumption	+	↑	↓	X
3	$\neg \text{pes}(x) \vee \neg \text{šteká}(x) \vee \neg \text{hryzie}(x)$	✓	Assumption	+	↑	↓	X
4	$\text{pes}(\text{Dunčo})$	✓	Assumption	+	↑	↓	X
5	$\neg \text{pes}(x) \vee \neg \text{pes}(x) \vee \neg \text{šteká}(x)$	✓	Resolution	+	↑	↓	X
2	<input type="text"/>			3	<input type="text"/>	<input type="text"/>	
6	$\neg \text{pes}(x) \vee \neg \text{šteká}(x)$	✓	Factoring	+	↑	↓	X
5	<input type="text"/>				<input type="text"/>		
7	$\neg \text{pes}(x) \vee \neg \text{pes}(x)$	✓	Resolution	+	↑	↓	X
1	<input type="text"/>			6	<input type="text"/>	<input type="text"/>	
8	$\neg \text{pes}(x)$	✓	Factoring	+	↑	↓	X
7	<input type="text"/>				<input type="text"/>		
9	$\square$	✓	Resolution	+	↑	↓	X
8	<input type="text"/>			4	<input type="text"/>	$x \rightarrow \text{Dunčo}$	

# Tableau editor

- Elm

Basic propositional | Prettify formulas | Print | Export as JSON | Import from JSON | Undo | Redo

(1)  $T(\text{kino}(\text{Fero}, \text{Anka}) \vee (\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \vee \text{hra}(\text{Fero}, \text{FeroVaPS})))$   $S^*$  [ 1 ] ...

(2)  $T(\text{kapela}(\text{PinkFloyd}) \wedge \text{hraciaKonzola}(\text{FeroVaPS}))$   $S^*$  [ 1 ] ...

(3)  $T(\neg \text{frustrovany}(\text{Fero}) \rightarrow \text{kino}(\text{Fero}, \text{Anka}))$   $S^*$  [ 1 ] ...

(4)  $T(\text{frustrovany}(\text{Fero}) \rightarrow (\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \vee \text{hra}(\text{Fero}, \text{FeroVaPS})))$   $S^*$  [ 1 ] ...

(5)  $T(\neg \text{kino}(\text{Fero}, \text{Anka}) \wedge (\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \wedge \text{hra}(\text{Fero}, \text{FeroVaPS})))$   $S^*$  [ 1 ] ...

(6)  $T(\text{hra}(\text{Fero}, \text{FeroVaPS}) \rightarrow \text{pocuva}(\text{Fero}, \text{PinkFloyd}))$   $S^*$  [ 1 ] ...

(7)  $T(\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \rightarrow \neg \text{frustrovany}(\text{Fero}))$   $S^*$  [ 1 ] ...

(8)  $F(\neg \text{hra}(\text{Fero}, \text{FeroVaPS}) \rightarrow \text{kino}(\text{Fero}, \text{Anka}))$   $S^*$  [ 1 ] ...

(9)  $T \rightarrow \text{hra}(\text{Fero}, \text{FeroVaPS})$   $\alpha$  [ 8 ] ...

(10)  $F \text{kino}(\text{Fero}, \text{Anka})$   $\alpha$  [ 8 ] ...

(11)  $F \text{hra}(\text{Fero}, \text{FeroVaPS})$   $\alpha$  [ 9 ] ...

---

(12)  $T \text{kino}(\text{Fero}, \text{Anka})$   $\beta$  [ 1 ] ... \* 10 12 Open

(13)  $T(\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \vee \text{hra}(\text{Fero}, \text{FeroVaPS}))$   $\beta$  [ 1 ] ...

(14)  $T \text{pocuva}(\text{Fero}, \text{PinkFloyd})$   $\beta$  [ 13 ] ... Add assumption Add Delete Close

(15)  $F \text{pocuva}(\text{Fero}, \text{PinkFloyd})$   $\beta$  [ 7 ] ... \* 15 14 Open

(16)  $T \neg \text{frustrovany}(\text{Fero})$   $\beta$  [ 7 ] ... Add assumption Add Delete Close

(17)  $T \text{hra}(\text{Fero}, \text{FeroVaPS})$   $\beta$  [ 13 ] ...

This tableau does not prove:

$(\text{kino}(\text{Fero}, \text{Anka}) \vee (\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \vee \text{hra}(\text{Fero}, \text{FeroVaPS}))) , (\text{kapela}(\text{PinkFloyd}) \wedge \text{hraciaKonzola}(\text{FeroVaPS})) , (\neg \text{frustrovany}(\text{Fero}) \rightarrow \text{kino}(\text{Fero}, \text{Anka})) , (\text{frustrovany}(\text{Fero}) \rightarrow (\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \vee \text{hra}(\text{Fero}, \text{FeroVaPS}))) , (\neg \text{kino}(\text{Fero}, \text{Anka}) \wedge (\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \wedge \text{hra}(\text{Fero}, \text{FeroVaPS}))) , (\text{hra}(\text{Fero}, \text{FeroVaPS}) \rightarrow \text{pocuva}(\text{Fero}, \text{PinkFloyd})) , (\text{pocuva}(\text{Fero}, \text{PinkFloyd}) \rightarrow \neg \text{frustrovany}(\text{Fero})) \vdash (\neg \text{hra}(\text{Fero}, \text{FeroVaPS}) \rightarrow \text{kino}(\text{Fero}, \text{Anka}))$

## Help

### Propositional and first-order logical symbols

Logical symbol	Symbols	Restrictions
Negation	$\neg, \sim, \bar{\phantom{x}}$	unary
Equality	$=, \neq$	binary, takes two terms
Inequality	$! =, / =, \neq$	
Conjunction	$\&, \wedge, \wedge$	
Disjunction	$ , \vee, \vee$	
Implication	$\rightarrow, \rightarrow$	strictly binary, must be parenthesized
Existence	$\exists$	

### Important notes

Note	Example
Each node contains a signed formula, i.e. it must be prefixed by T or F.	$T \forall \text{forall } x P(x)$ $F \exists x \forall y (K(x,y) \wedge G(y,x))$
Write each premise/assumption and conclusion/goal with no references. Sign premises with T and sign conclusions with F.	(1) $T(A \rightarrow B)$ [] (2) $F \neg(A \wedge \neg B)$ []
When substituting a variable x with a term t, t must not contain any variable which is bound at any occurrence of x.	<b>Incorrect example:</b> (1) $T \forall x \exists y P(x,y)$ [] (2) $T \exists y P(t(y),y)$ [ $x \rightarrow t(y)$ ] []

# Prieskumník štruktúry logiky

- JavaScript
- Typescript
- React, Redux
- Bootstrap 4

{ } Množiny Graf Exportovať Importovať

**Jazyk  $\mathcal{L}$**  ✓ Hotovo ?

Individuové konštanty  
 $C_{\mathcal{L}} = \{ \text{Curo, Filip, Felix, Granule, Kosti, Kuracie\_prsia} \}$

Predikátové symboly  
 $\mathcal{P}_{\mathcal{L}} = \{ \text{kocur/1, ma\_rad/2, zozral/2} \}$

Funkčné symboly  
 $\mathcal{F}_{\mathcal{L}} = \{ \text{fce1/1} \}$

**Štruktúra  $\mathcal{M} = (D, i)$**  ?

Doména  
 $D = \{ \text{c, f, g, k, kp} \}$

Interpretácia individuových konštánt

$i(\text{Curo}) = \text{c}$

$i(\text{Filip}) = \text{f}$

$i(\text{Kosti}) = \text{k}$

$i(\text{Kuracie\_prsia}) = \text{kp}$

$i(\text{Granule}) = \text{g}$

**Pravdivosť formúl v štruktúre  $\mathcal{M}$**  ?

$\varphi_1$   $\exists y \text{ kocur}(y)$  ✓ Správne

$\mathcal{M} \models \varphi_1[e]$

$\varphi_2$   $\exists j (\exists z (\neg \text{ma\_rad}(z, j) \wedge \text{zozral}(z, j)))$  ✗ Nesprávne

$\mathcal{M} \models \varphi_2[e]$

+ Pridať

**Hodnoty termov v  $\mathcal{M}$**  ?

$r_1$  Kosti ✓ Správne

$\mathcal{M} \models k$

$r_2$  fce1(Curo) ✓ Správne

$\mathcal{M} \models c$

+ Pridať

# Integrácia existujúcich aplikácií

- Prerobenie existujúcich aplikácií na knižnice s Reactovským komponentom
- Každá aplikácia implementuje navrhnuté rozhranie

```
1  interface AppProps {  
2    instance: any,  
3    isEdited: boolean,  
4    onChange: () => void  
5  }  
6  
7  interface PrepareResult {  
8    instance: any,  
9    getState: (instance: any) => any,  
10 }  
11  
12 export function prepare(initialState: any): PrepareResult  
13 export function AppComponent(props: AppProps): JSX.Element
```

# Integrácia s Githubom

- Autentifikácia
- Prehliadanie zadaní v repozitáry
- Vytváranie nových hárkov
- Automatické ukladanie hárku

# Testovanie

- Dostali sme spätnú väzbu od 12 študentov
- Pozitívne hodnotenie
- Návrhy na zlepšenie
- Odhalili sa niektoré chyby a problémy

**Ako sa vám v hárku pracovalo?** Velmi dobre, je to prehľadne, a jednoducho sa v hárku pracuje.

**Čo sa vám na pracovnom hárku páči?** Všetko je na jednom mieste, automaticky sa ukladajú zmeny.

**Čo sa vám na pracovnom hárku nepáči, čo by ste zmenili, vylepšili, prípadne ako?** Ani neviem.

**Koľkými hviezdami (1-5) hodnotíte túto aplikáciu?**

★★★★★

**Ako sa vám v hárku pracovalo?**

super, zadanie hneď nad odpoveďami, ukladá sa to rovno na github. parada

**Čo sa vám na pracovnom hárku páči?**

je super, že je všetko takto pokope

**Čo sa vám na pracovnom hárku nepáči, čo by ste zmenili, vylepšili, prípadne ako?**

niekto to aj písal do chatu na teams, nemuselo by to ziadat o všetky SUKROMNE repositare z githubu... staci ten jeden

**Koľkými hviezdami (1-5) hodnotíte túto aplikáciu?**

★★★★★

# Záver

- Vytvorili sme funkčnú a použiteľnú aplikáciu
- Použitý nápad a implementácia boli pozitívne ocenené študentami v spätnej väzbe
- Aplikácia sa plánuje nasadiť na vyučovaní v budúcom akademickom roku
- Možnosti vylepšenia aplikácie

Ďakujem za pozornosť