

Znalostní Systém pro Řešení Logických Úloh

Autor: Martin Riedl

Email: marty69@students.zcu.cz

Datum: 09.12.2024

Verze: 1.0

Obsah

1. Úvod
 2. Struktura Projektu
 3. Popis Modulů
 - 1. riedl_main.py
 - 2. riedl_houses.py
 - 3. riedl_constraints.py
 - 4. riedl_solver.py
 4. Instalace a Spuštění
 5. Použití
 6. Příklady Výstupu
 7. Podpora a Kontakt
-

Úvod

Tento projekt představuje **Znalostní Systém pro Řešení Logických Úloh**, konkrétně logickou úlohu známou jako "Zebra Puzzle". Systém je navržen pro určení správného přiřazení atributů (barva domu, národnost obyvatel, značky cigaret, nápojů a domácích zvířat) pro pět rodinných domů na základě daných omezení. Projekt je strukturován do několika modulů pro lepší organizaci a údržbu kódu.

Struktura Projektu

Projekt je rozdělen do následujících souborů:

1. **riedl_main.py** – Hlavní spustitelný soubor.
2. **riedl_houses.py** – Definice domů a atributů.
3. **riedl_constraints.py** – Definice validačních funkcí.
4. **riedl_solver.py** – Implementace řešícího algoritmu.

Všechny soubory mají prefix **riedl_** a obsahují podrobnou dokumentaci v podobě docstringů.

Popis Modulů

1. riedl_main.py

Účel:

Hlavní spouštěcí soubor projektu.

Funkce:

- `main()`
 - Inicializuje strukturu domů pomocí funkce `initialize_houses()`.
 - Nastavuje počáteční známé atributy podle daných skutečností.
 - Specifikuje pořadí atributů pro přiřazování.
 - Spouští řešící algoritmus pomocí funkce `solve()`.
 - Zobrazuje výslednou tabulku s přiřazenými atributy pro každý dům.
 - Určuje vlastníka zebry a osobu, která pije vodu.

Importy:

- `initialize_houses` z `riedl_houses.py`
- `solve` z `riedl_solver.py`
- `is_valid` z `riedl_constraints.py`

2. riedl_houses.py**Účel:**

Definuje strukturu domů a možné hodnoty pro každý atribut.

Funkce:

- `initialize_houses()`
 - Vytváří seznam pěti domů, každý reprezentovaný jako slovník s atributy nastavenými na `None`.

Definované Atributy:

- **Barva:** červený, zelený, bílý, žlutý, modrý
- **Národnost:** Angličan, Španěl, Norek, Ukrajinec, Japonec
- **Cigarety:** Sparty, Chesterfield, Winston, Lucky-Strike, Parliaments
- **Nápoj:** čaj, káva, mléko, pomerančový džus, voda
- **Domácí zvíře:** pes, liška, sněhy, koně, zebra

3. riedl_constraints.py**Účel:**

Obsahuje funkci pro validaci aktuálního přiřazení atributů podle daných omezení.

Funkce:

- `is_valid(houses)`
 - Kontroluje, zda aktuální přiřazení atributů splňuje všechna definovaná omezení.
 - Vrací `True`, pokud všechna omezení jsou splněna, jinak `False`.

Omezení Implementovaná ve Funkci:

1. Angličan žije v červeném domě.
2. Španěl má psa.
3. Norek žije v prvním domě zleva.
4. Obyvatel žlutého domu kouří Sparty.

5. Kuřák Chesterfieldů žije vedle majitele lišky.
6. Norek žije v modrém domě.
7. Kuřák Winstonů pěstuje sněhy.
8. Kuřák Lucky-Strike pije pomerančový džus.
9. Ukrajinec pije čaj.
10. Japonci kouří Parliaments.
11. Majitel koně žije vedle kuřáka Sparty.
12. Majitel zeleného domu pije kávu.
13. Zelený dům je bezprostředně napravo od bílého domu.
14. Ve středním domě se pije mléko.

4. riedl_solver.py

Účel:

Implementuje řešící algoritmus pomocí rekurzivního backtrackingu pro přiřazení atributů domů.

Funkce:

- **solve(houses, attributes, is_valid_func)**
 - Rekurzivně přiřazuje atributy domům podle zadaného pořadí.
 - Kontroluje jedinečnost hodnot pro každý atribut.
 - Používá funkci **is_valid_func** pro validaci aktuálního přiřazení.
 - Vrací řešení jako seznam domů, pokud je nalezeno, jinak **None**.
- **find_next_attribute(houses, attributes)**
 - Najde další atribut a index domu, který potřebuje přiřazení.
 - Vrací tuple (název atributu, index domu) nebo (None, None) pokud je vše přiřazeno.
- **get_possible_values(attribute, houses)**
 - Vrací možné hodnoty pro daný atribut, které ještě nebyly přiřazeny.
- **is_unique(attribute, value, houses)**
 - Kontroluje, zda je hodnota pro daný atribut jedinečná mezi všemi domy.

Instalace a Spuštění

1. Uložení Souborů:

- Ujistěte se, že všechny čtyři soubory (**riedl_main.py**, **riedl_houses.py**, **riedl_constraints.py**, **riedl_solver.py**) jsou uloženy ve stejném adresáři.

2. Spuštění:

- Otevřete terminál nebo příkazový řádek.
- Přejděte do adresáře, kde jsou soubory uloženy.
- Spustěte program pomocí příkazu:

```
python riedl_main.py
```

Použití

Po spuštění programu se systém pokusí vyřešit logickou úlohu a zobrazí výslednou tabulku s přiřazenými atributy pro každý dům. Dále systém odpoví na otázky týkající se vlastníka zebry a osoby, která pije vodu.

Příklady Výstupu

Pozice | Barva | Národnost | Cigarety | Nápoj | Domácí zvíře

1		žlutý		Norek		Sparty		voda		koně
2		modrý		Ukrajinec		Chesterfield		čaj		liška
3		červený		Angličan		Winston		mléko		sněhy
4		zelený		Španěl		Lucky-Strike		pomerančový džus		pes
5		bílý		Japonec		Parliaments		káva		zebra

Kdo vlastní zebra? Japonec

Kdo pije vodu? Norek

Podpora a Kontakt

Pokud narazíte na jakékoliv problémy nebo máte dotazy týkající se tohoto systému, neváhejte kontaktovat autora:

Martin Riedl Email: marty69@students.zcu.cz