

# ИНТЕРАКТИВНА ПЛАТФОРМА ЗА ОБУЧЕНИЕ И ОЦЕНЯВАНЕ НА УЧЕНИЦИ ПО ЛОГИЧЕСКО ПРОГРАМИРАНЕ С ПРОЛОГ: АВТОМАТИЗИРАНО СЪЗДАВАНЕ НА БАЗИ ЗНАНИЯ И ПРАВИЛА

Магдалена Мъглижанова, Венета Табакова-Комсалова,  
Станимир Стоянов, Ласка Костадинова-Цанкова

*Резюме.* В тази статия разглеждаме възможността за въвеждане на обучение по логическо програмиране на езика Пролог за ученици в средното училище. Обучението се осъществява по интерактивен и иновативен начин чрез обединението на SQL и Пролог в уеб приложение. Приложението улеснява преподавателите по програмиране, като им предоставя инструмент, който им позволява да демонстрират принципите на логическото програмиране и работата с бази данни, интегрирани в рамките на ASP.NET. Благодарение на интеграцията между SQL и Пролог, учениците имат възможност да експериментират с практически примери, съчетаващи логика и данни, което допринася за по-дълбоко разбиране на абстрактните концепции в програмирането.

**Ключови думи:** обучение, изкуствен интелект, логическо програмиране, проектно-базирано обучение, STEM.

## Въведение

С напредъка на изкуствения интелект (ИИ) логическото програмиране придобива все по-голямо значение в образованието на учениците по целия свят. Въпреки това, в гимназиите, логическото програмиране не е масово застъпено, а потенциалът му остава неразкрит за голяма част от обучаващите се. Логическото направление в изкуствения интелект е един от първите и най-фундаментални подходи за създаване на интелигентни системи. То се основава на идеята, че мисленето може да бъде формализирано в логически изрази, а решаването на проблеми може да се свежда до манипулиране на тези изрази по строго определени правила. Един от основните принципи е формализация на знанията. Логическите системи

използват формални езици за представяне на факти, правила и отношения между обектите в света. Това позволява на машините да разбират и манипулират информация по прецизен и систематичен начин.

В този контекст нашият екип създаде платформа за обучение на езика за логическо програмиране Пролог. Нашата цел е да улесним учениците в усвояването на основите на програмирането, като им предоставим лесен и достъпен начин за учене. Платформата не само ще осигури интерактивна среда за изучаване на Пролог, но и ще демонстрира как знанията и информацията могат да бъдат съхранявани и използвани ефективно чрез интеграция между SQL и логически езици [1]. За първи път учениците ще имат възможността да изучават програмиране, като взаимодействат с реални логически бази от знания и автоматизирани системи за оценка.

Програмният език Пролог предлага уникални предимства, особено в сферата на изкуствения интелект и логическото програмиране [2, 3]. В отговор на това създаваме платформа, която предлага интерактивни уроци, мотивиращи учениците да научат този програмен език по забавен и ангажиращ начин. Всеки потребител може да създаде собствен профил и да започне обучението си, което осигурява индивидуален подход. Учителите също ще разполагат с инструменти за проследяване на напредъка на учениците, което им позволява да наблюдават развитието на техните компетенции.

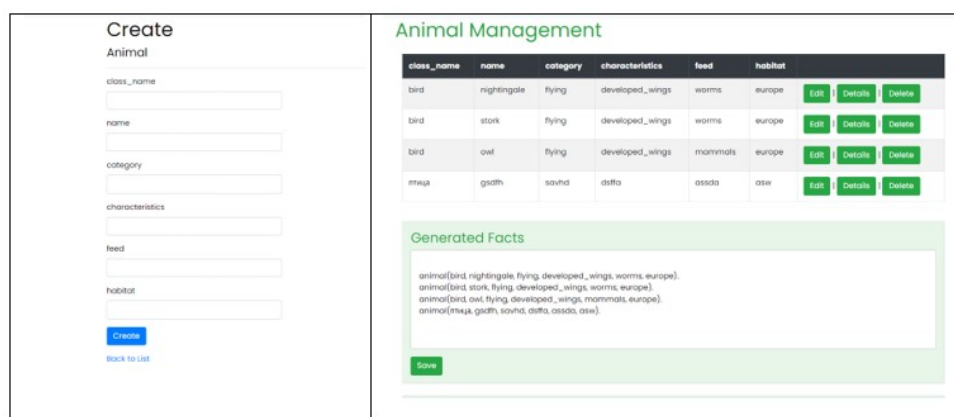
### **Функционалност на приложението**

Системата, която разработихме, позволява на учениците да създават, съхраняват и изпълняват факти и правила в среда, базирана на езикът за логическо програмиране Пролог и стандартен език за съхраняване, манипулиране и извличане на данни в бази данни SQL. Взаимодействието между логическите правила и фактите е основната цел на системата, като учениците могат да създават сложни логически запитвания, които се изпълняват и анализират автоматично от приложението (Фиг. 1). Фактите и правилата, се съхраняват като записи в базата данни и могат да бъдат използвани по-късно. Целта е да се усвои логическото програмиране и да се анализират отговорите на системата при различна обработка на данните, които могат да се генерират под формата на факти.

Като начало е създадена база от знания за животни и растения, тъй

като през изминалата учебна година в ПГЕЕ – гр. Пловдив учениците от специалност „Приложно програмиране“ изучават за първи път езика за програмиране Пролог и първоначалните ресурси са свързани с тази тематика.

Докато обучаващите се изграждат йерархия от класове животни, използвайки факти, те развиват следните умения, като логическо мислене, аналитични способности, умения за решаване на проблеми, работа в екип, умения за структуриране на информация, критично мислене, способност за работа с данни и факти [4, 5].



Фигура 1. Съхранение и управление на факти и правила

Приложението предоставя възможност за консултиране на съхранените факти с правила, като това се извършва в Пролог среда, разположена в отделен клас `PrologService`. Това дава на учениците възможността да видят в действие принципите на дедуктивното логическо програмиране и как фактите и правилата се свързват помежду си.

**Интегриране на Пролог в ASP.NET.** Платформата интегрира език за програмиране Пролог в среда на ASP.NET. По този начин съчетава възможностите на Пролог с инфраструктурата на SQL. Интегрирането на Пролог в ASP.NET се осъществява благодарение на `SwiPLCs` библиотеката. Тази интеграция предоставя възможност на приложението да зарежда факти и правила в Пролог, да изпълнява заявки и да връща резултати.

Процесът на интеграция се осъществява като приложението изпраща създадените факти и правила към Пролог, чрез метода `ConsultFromString()`, който зарежда логически изрази и факти в Пролог. След това се използват Пролог запитвания, за да се извлече информация или да се

направи оценка на правилата, съхранени в базата данни.

**Генериране на факти от таблицата в базата данни.** Фактите в Пролог описват същности и техните свойства. В приложението тези факти се генерират автоматично от таблицата с данни за животни в SQL базата данни. Например, таблицата съдържа информация за животни (като клас, име, категория, хранене, местообитание и т.н.), която се използва за създаването на факти от типа:

```
animal(bird, swallow, flying, developed_wings, worms, europe).  
animal(mammal, lion, predator, carnivore, meat, africa).
```

Процесът на създаване на факти включва извличане на данни от SQL базата данни, формиране на факти и записването им в низ, който след това се консултира от Пролог. Това се осъществява чрез метода `GetAllAnimals()` в приложението, който извлича информацията от базата и съставя низ със съответните Пролог факти.

**Автоматизирано създаване на файл с работата на учениците.** Приложението позволява автоматизирано генериране на отчет за работата на учениците, което се съхранява в текстови файлове с разширение „.pl“. Това включва както фактите, така и правилата, използвани от учениците.

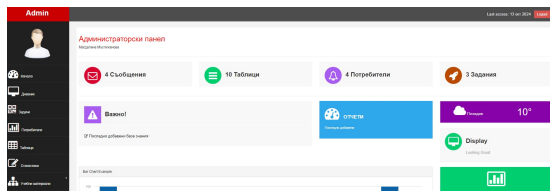
Методът `SaveFactsToFile(string fileName)` създава файл, в който се записват всички факти, генерирани от данните в базата, заедно с правилата. По този начин учителите могат лесно да проследят какви данни и правила са били използвани в оценката на учениците.

**Администраторски и потребителски интерфейс.** Администраторският профил е предназначен за учителите и предоставя набор от различни функционалности, които улесняват управлението на потребителите и ресурсите в платформата.

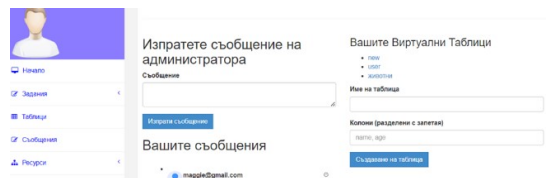
*Функционалности на администраторския профил (Фиг. 2) са:*

- Управление на потребителски профили;
- Контрол на достъпа;
- Наблюдение на активността;
- Управление на съобщения;
- Отчети и анализи;

- Поддръжка.



Фигура 2. Административен панел



Фигура 3. Потребителски панел

*Функционалности на потребителския профил (Фиг. 3).* Потребителите могат да създават свои виртуални таблици, които да запълват с факти и правила. Виртуалните таблици позволяват на потребителите да създават и управляват свои собствени набори от данни, които са персонални за всеки потребител. Те също така, предлагат възможност за динамично добавяне и модификация на структури от данни без необходимост от пренаписване на кода или манипулиране на базата данни. Например могат да добавят виртуални таблици, за да съхраняват данни, без да е необходимо да променят структурата на основната база данни. Виртуалните таблици могат да помогнат за подобряване на сигурността на данните, като се позволява на потребителите да работят само с данните, които имат право да виждат. Членовете на екип могат да добавят свои таблици и данни, които да споделят помежду си, което подобрява комуникацията и синхронизацията на информацията. Също така потребителите могат да изпращат съобщения до администраторите, които след това могат да бъдат прегледани и управлявани.

*Добавяне на правила в базата данни.* Приложението позволява лесно добавяне на нови правила в системата чрез потребителския интерфейс. В таблицата Rules на SQL базата данни се съхраняват всички правила на Prolog, които след това се зареждат автоматично в логическата база знания. Потребителите (например, учители) могат да добавят нови логически правила чрез специална страница, която записва правилото в базата данни. След добавянето, правилата се зареждат и използват за оценка в приложението. Например, добавено правило може да бъде нещо подобно на:

```
is_mammal(Name) :- animal(mammal, Name, _, _, _, _).
```

Така, при изпълнение на запитвания към Пролог, тези нови правила се използват за логическо заключение.

### Основни етапи при обучаване и оценяване

При обучението по логическо програмиране и работа с бази знания, оценяването на учениците преминава през няколко основни етапа. Всеки етап е внимателно структуриран, за да развие постепенно уменията на учениците и да им помогне да разбират все по-сложни концепции. Основната идея е да се комбинира теоретичното разбиране с практическа работа, като всеки етап (Таблица 1) от обучението се оценява от учителя.

Таблица 1. Основни етапи при обучаване и оценяване

Етап	Ученик	Учител
Създаване на мисловни карти	Мисловните карти служат като първа стъпка за изграждане на логическа структура, която по-късно ще бъде превърната в база знания.	Учителите оценяват мисловните карти на учениците въз основа на тяхната логическа последователност, яснота и цялостното разбиране на основните категории и техните взаимовръзки.
Създаване на база знания	Следващият етап е създаването на база знания, която включва факти и основни твърдения. Учениците превръщат концепциите от мисловните карти в конкретни факти, които могат да бъдат обработени от логическа система като Пролог.	На този етап учителят оценява правилността на създадените факти, тяхната релевантност и пълнота. Учениците получават обратна връзка за корекции или подобрения преди да преминат към следващия етап.
Генериране на факти и проверка	Важна част от процеса е създаването на факти чрез приложението, което учениците са разработили. Те извличат фактите от базата данни, която са създали, и ги представят в подходящ формат, който може да бъде оценен. Тези факти се използват като основа за логическите правила, които следват.	Получените факти се оценяват от учителя за тяхната адекватност спрямо изискванията на задачата.

Добавяне на правила	След като учениците са създали фактите, следващият етап е добавянето на логически правила. Тези правила обобщават фактите и извършват логически изводи върху тях. Важна характеристика на правилата е, че комуникират помежду си и надграждат съществуващата база знания. Добавянето на всяко ново правило помага на учениците да надградят своите логически умения и способността да мислят абстрактно. Това стимулира обучението им, тъй като всяко следващо правило изисква по-задълбочено разбиране на предишните.	Учителят оценява правилността и сложността на създадените правила, както и уменията на учениците да съчетават различни правила и факти в по-сложни логически изводи.
Общи и индивидуални бази знания	Една от уникалните характеристики на приложението е, че то дава възможност за създаване както на обща база знания, така и на индивидуални бази знания за всеки ученик. Общата база знания, например в сферата на животни и растения, ще бъде попълвана съвместно от много ученици. Това насърчава сътрудничеството между учениците, като те изграждат обща система от факти и правила, която расте и се развива с участието на всички. В същото време, всеки ученик има възможността да създава и собствена, индивидуална база знания, която отразява неговите лични знания и разбиране по темата. Това позволява на учениците да развиват самостоятелно мислене и умения за анализ, докато същевременно използват ресурсите на общата база знания.	Учителите оценяват индивидуалните бази знания на всеки ученик въз основа на тяхната оригиналност, сложност и степента на разбиране на Пролог.

Финална оценка и напредък	В края на процеса учителите правят финална оценка на целия проект, като вземат предвид всички етапи – от мисловните карти до създаването на базата знания и правилата. Оценяването включва анализ на това как учениците са се справили с прогресивното изграждане на знания, както и как са интегрирали и използвали фактите и правилата в своята система. Учителите могат лесно да оценяват учениците чрез автоматизираната система за оценка, която се базира на коректността на създадените от учениците правила и факти, както и на резултатите от изпълнените запитвания.	Учителите могат да задават различни Пролог запитвания, за да проверят знанията на учениците. Приложението изпълнява тези запитвания чрез метода <code>ApplyRulesAndGetResults(string query)</code> , като извлича резултатите и ги показва на уеб интерфейса. Също така, могат да се проверят файловете, които са изпратили учениците.
---------------------------	--	--

### Предимства на приложението

С така създаденото приложение се улеснява обучението по логическо програмиране, като се интегрира езикът за логическо програмиране Пролог и стандартен език за съхраняване, манипулиране и извличане на данни в бази данни SQL. Учениците могат лесно да разберат как логическото програмиране работи в реални проекти, свързани с бази от данни. Автоматизираната система помага на учителите по програмиране да оценяват по-ефективно и да предоставят персонализирана обратна връзка. Също така, позволява на учителите автоматично да оценяват резултатите на учениците въз основа на логически запитвания и правила, без да е необходимо ръчно проверяване.

### Заклучение

Приложението предоставя удобен начин за обучение по логическо програмиране и е иновация в системата за оценяване на ученици в гимназиите. Чрез интеграцията на Пролог и SQL, системата предлага автоматизирано съхранение и изпълнение на факти и правила, което улеснява както учениците, така и учителите. По този начин учениците моделират знания от различни области, вкл. българския фолклор. Въвеждането на логическо програмиране в училищата ще развие критично мислене и аналитични умения у учениците и ще ги подготви за бъдещето в областта на изкуствения интелект и технологиите.



## Благодарности

Авторите изказват благодарност към научен проект КП-06-М62/2 „Моделиране на знания в областта на българския фолклор“, финансиран по Фонд „Научни изследвания“, за финансиране на настоящата работа.

## Литература

- [1] E. Skvortsov, Y. Xia, S. Bowers, B. Ludäscher, From Logic Programming to Programming in Logica: A First-Course in Declarative Data Science and Engineering, *Workshop Proceedings of the 40th International Conference on Logic Programming (ICLP-WS 2024)*, PEG 2024: 2nd Workshop on Prolog Education, October, 2024, Dallas, USA, <https://ceur-ws.org/Vol-3799/paper6PEG2.0.pdf>.
- [2] M. Genesereth, General Game Playing – Killer App for Logic Programming, *Workshop Proceedings of the 40th International Conference on Logic Programming (ICLP-WS 2024)*, PEG 2024: 2nd Workshop on Prolog Education, October, 2024, Dallas, USA, <https://ceur-ws.org/Vol-3799/short3PEG2.0.pdf>.
- [3] S. Villarroe, C. Gimenez, J. Rodríguez, L. Cecchi, Democratising Access to Logic Programming. A Web Application Design Tool for Querying Prolog Code, *Workshop Proceedings of the 40th International Conference on Logic Programming (ICLP-WS 2024)*, PEG 2024: 2nd Workshop on Prolog Education, October, 2024, Dallas, USA, <https://ceur-ws.org/Vol-3799/paper5PEG2.0.pdf>.
- [4] V. Tabakova-Komsalova, A. Dukovski, L. Kostadinova-Tsankova, M. Maglizhanova, Artificial Intelligence And Logic Programming In Stem Education, *Education and Technologies*, Vol. 14, Issue 1, 2023, pp. 193–199, ISSN: 1314-1791 (print), ISSN: 2535-1214 (online), DOI: <https://doi.org/10.26883/2010.231.5069>.
- [5] M. Maglizhanova, V. Tabakova-Komsalova, S. Stoyanov, L. Kostadinova-Tsankova, Aproject-Based Approach To Introducing Logic Programming In The Secondary School, *Education and Technologies*, Vol. 15, Issue 1, 2024, pp. 183–189, ISSN: 1314-1791 (print), ISSN: 2535-1214 (online), DOI: <https://doi.org/10.26883/2010.231.5063>.

Магдалена Мъглижанова<sup>1</sup>, Венета Табакова-Комсалова<sup>1,2</sup>,  
Станислав Стоянов<sup>1,2</sup>, Ласка Костадинова-Цанкова<sup>1</sup>,

<sup>1</sup> Пловдивски университет „Паисий Хилендарски“

Факултет по математика и информатика,

<sup>2</sup> Българска академия на науките

Институт по информационни и комуникационни технологии,

Автор за кореспонденция: v.komsalova@uni-plovdiv.bg

## INTERACTIVE PLATFORM FOR TEACHING AND ASSESSING STUDENTS IN LOGIC PROGRAMMING WITH PROLOG: AUTOMATED CREATION OF KNOWLEDGE BASES AND RULES

Magdalena Maglizhanova, Veneta Tabakova-Komsalova,  
Stanimir Stoyanov, Laska Kostadinova-Tzankova

**Abstract.** *In this article, we consider the possibility of introducing logic programming training in the Prolog language to secondary school students. The training is carried out in an interactive and innovative way by combining SQL and Prolog in a web application. The application facilitates programming educators by providing them with a tool that allows them to demonstrate the principles of logic programming and working with databases integrated within ASP.NET. Thanks to the integration between SQL and Prolog, students have the opportunity to experiment with practical examples combining logic and data, which contributes to a deeper understanding of abstract concepts in programming.*

**Key words:** Education, Artificial Intelligence, Project-Based Learning, Logic Programming, STEM.

### Acknowledgments

The research is supported by the project KP-06-M62/2 “Modelling of knowledge in the field of Bulgarian folklore” funded by the National Research Fund.