

ДИФЕРЕНЦИРАН ПОДХОД ЗА ИЗУЧАВАНЕ НА ИЗКУСТВЕН ИНТЕЛЕКТ В СРЕДНОТО УЧИЛИЩЕ

¹ Венета Веселинова Табакова-Комсалова, ² Магдалена Иванова Мъглижанова,

¹ Станимир Недялков Стоянов

¹ Пловдивски университет „Паисий Хилендарски“, Пловдив, ИИКТ, БАН

² Пловдивски университет „Паисий Хилендарски“, Пловдив

A DIFFERENTIATED APPROACH TO LEARNING AI IN SECONDARY SCHOOL

¹ Veneta Veselinova Tabakova-Komsalova, ² Magdalena Ivanova Maglizhanova,

¹ Stanimir Nedyalkov Stoyanov

¹ Plovdiv University „Paisii Hilendarski“, Plovdiv, ICT BAS, Bulgaria

² Plovdiv University „Paisii Hilendarski“, Plovdiv, Bulgaria

Abstract: Evolutionarily, artificial intelligence is developing in two independent directions – symbolic and subsymbolic artificial intelligence. In the mid-1980s, people started talking about so-called in-between models that embody the idea of hybrid intelligent systems. Recently, we have seen an increased interest in such models, now under the notion of integrated artificial intelligence.

This paper presents a differentiated approach to teaching and learning AI in secondary school, distinguishing symbolic from subsymbolic AI. The motivation for using such an approach is mainly rooted in the different theoretical foundations of the two fields – in the case of symbolic artificial intelligence, it is formal logic, and as far as subsymbolic artificial intelligence different concepts from linear algebra, probability theory and statistics. In this sense, an interesting question is what depth of theoretical background is assumed for students to be able to create and implement (not just use) AI systems. This paper attempts to explore this issue based on two examples. The first example is a knowledge-based system delivering subject knowledge across disciplines. The second example is using a neural network to identify unhealthy ingredients in food products.

Keywords: *symbolic AI, subsymbolic AI, integrated AI, education, STEM, logic programming, neural networks*

Въведение. Изкуственият интелект (ИИ) вече е реалност с влияние в икономиката, здравеопазването и образованието, но в средното училище интеграцията му е ограничена до основно запознаване. Настоящата статия предлага диференциран подход в обучението, разграничавайки символния ИИ (базиран на логика и символи) от подсимволния ИИ (машинно обучение и невронни мрежи, основани на линейна алгебра и статистика).

Нашият подход цели адаптирано обучение, което отговаря на различните способности и интереси на учениците, от базови програмни концепции до сложни интердисциплинарни приложения. Ключов въпрос е какво ниво на подготовка е нужно, за да станат учениците не само потребители, но и създатели на ИИ решения.

В статията се разглеждат два примера: база знания, илюстрираща символния ИИ, и приложение с невронна мрежа за разпознаване на нездравословни съставки, демонстриращо подсимволния ИИ. Анализират се предизвикателствата при интеграцията на ИИ в училищната програма, като се отчитат различни учебни условия, цели и нивото на подготовка на учители и ученици.

Символен, подсимволен и интегриран изкуствен интелект. Исторически ИИ еволюира в две основни направления – символен и подсимволен (статистически) ИИ. Символният ИИ използва логически конструкции и продукционни системи като Пролог и експертни системи, подходящи за ясно дефинирани задачи с дедукция [1]. Той обаче е ограничен при работа с несигурни и неструктурирани данни. Подсимволният ИИ включва невронни мрежи и вероятностни модели, базирани на линейна алгебра и статистика [2], които позволяват машинно обучение и дълбоко учене, но са „черна кутия“ с ниска прозрачност. Двете парадигми често се конкурират – периоди на „зима“ в едното предизвикват интерес към другото, а „летата“ фокусират вниманието върху успешния подход. Това напрежение има философски корени, като дебатът между рационализма и емпиризма, илюстриран от Максвел [3], Айнщайн и Бор [4], и съвременния спор между Чомски и Норвиг [5]. В човешкото познание дедукцията и индукцията са взаимносвързани, а разделянето на символно и подсимволно знание е трудно [6]. Например, автоматизираното доказване на те-

ореми е традиционно в символния ИИ, но дълбокото учене също решава такива задачи [7].

Напоследък се увеличава интересът към хибридни системи, които комбинират силните страни на двете парадигми чрез междинни методи, преодоляващи пропастта между тях [8], [9]. Тези системи съчетават интерпретируемостта на символния ИИ с адаптивността на машинното обучение, което подпомага автономни решения и намалява зависимостта от големи данни. Марвин Мински подчертава, че нито чисто числените, нито чисто символните системи сами по себе си са достатъчни, и предлага мащабни архитектури, комбиниращи различни представяния за преодоляване на недостатъците им [10]. Според [11], прогресът в ИИ изисква интеграция на символни и статистически методи за по-ефективно и обяснимо машинно учене, като дедукцията и индукцията ще останат основни двигатели на бъдещото развитие.

Диференциран подход към преподаването на ИИ в училище. Въвеждането на ИИ в средното образование е необходимо, но изисква внимателно структуриране, за да отговори на разнородните интереси и способности на учениците. За да бъде ефективно, такова обучение трябва да отчита следните ключови аспекти:

- Диференциация според направленията на ИИ. ИИ е широка научна област с множество подходи, подобно на математиката, която се разделя на алгебра, геометрия, анализ и др. Затова е целесъобразно, изучаването му в училище да включва: символен ИИ (базиран на логика, правила и знания) и подсимволен ИИ (машинно учене, невронни мрежи).
- Различна теоретична основа – различни педагогически подходи. Символният ИИ използва формална логика, лингвистични модели и алгоритмично мислене, което го прави по-достъпен за всички ученици и особено тези, които са с хуманитарна насоченост. Подсимволният ИИ разчита на вероятности и статистика, линейна алгебра и оптимизационни методи, затова е по-подходящ за ученици със силни математически умения. Училищната програма трябва да предлага както базови знания за двата подхода, така и възможност за по-задълбочено изучаване според интересите на учениците.
- Масово и елитарно обучение. Масово обучение, включващо основно обучение за работа и използване на интелигентни системи или само основни концепции на ИИ, които са достъпни за всички ученици, независимо от профила им (напр. какво е ИИ, етични въпроси, приложения в ежедневието). Удачно е да се въведе и логическото програмиране (Пролог) което използва логика и е лесно разбираемо дори и от по-малки ученици. Проучването на Sессhи и съавт. [12] показва, че логическото програмиране е работещ и желан подход още при деца на 8-10 години в две държавни училища в Аржентина. Елитарно обучение може да се провежда под формата на специализирани курсове или клубове за ученици с необходими предварителни познания и определени интереси в областта. Това осигурява баланс между широкообразователната цел на училището и подкрепата за талантиливи ученици.
- Образователна стойност и интердисциплинарност. Изучаването на ИИ не трябва да бъде изолирано, а да се свързва с други предмети. За хуманитаристи – връзка между символен ИИ и история, география, лингвистика, право или когнитивни науки. А за природоматематиките – приложение на подсимволен ИИ в биология, физика или икономика. Това прави обучението по-практично и мотивиращо за учениците.

Предлагаме подход за преподаване, който разделя ИИ на два ясно различими дяла, базирани на съответната им теоретична основа. Това позволява на учениците по-добре да осъзнаят какво стои зад „интелигентността“ на съвременните системи и как самите те могат да създават такива системи.

Демонстрационни примери. Чрез разработените и представени по-долу примери ние демонстрираме нашия подход. Използвано е проектно-базирано обучение в СТЕМ за въвеждане на ИИ в средното училище с логическо програмиране на Пролог и представяне основите на подсимволния ИИ чрез невронни мрежи.

- Базирана на знания система (символен ИИ)

В този пример се създава базирана на правила система, която доставя специализирани знания за пещерите в България. Първата стъпка е да се сформират екип от учители. Този екип подготвя темата на проекта, която в случая изисква интердисциплинарен подход за решаване, и поради което в него са включени учители по химия, физика, биология, история, география и информатика. Учителите определят също информационните източници.

Втората стъпка е сформирание на екип от ученици, спрямо техните интереси. Ученическият екип, със съдействие на учителите, на основата на анализа на информацията определя обектите и техните характеристики (химични и физични свойства, местоположение, възраст, трудност, обитатели, интересни факти и др.), които ще бъдат представени в базата знания.

Третата стъпка е създаване на базата знания, описващи различните пещери и техните характеристики в езика за логическо програмиране Пролог (фиг. 1).

```

/* cave (Име, Дължина, Местоположение, Възраст, Трудност) */
cave('Дяволското гърло', 1500, 'Смолян', 800000, 5).
cave('Леденика', 300, 'Враца', 450000, 2).
cave('Снежанка', 145, 'Пещера', 600000, 1).
cave('Магурата', 2500, 'Белоградчик', 900000, 3).
cave('Ухловица', 460, 'Смолян', 550000, 3).
cave('Бачо Киро', 1200, 'Дряново', 750000, 4).
cave('Ягодинска', 1050, 'Борино', 880000, 2).
...

```

Фигура 1. Сегмент от базата знания

На четвъртата стъпка учениците разширяват базата знания с представянния на класификации, *взаимовръзки и релации* между обектите (фиг. 2).

```

/* Изброява всички пещери с пълна информация */
list_caves :- forall(cave(Name, Length, Location, Age, Diff),
    format('~w: Length = ~w m, Location = ~w,
    Age = ~w years, Difficulty = ~w~n', Name, Length,
    Location, Age, Diff)).

/* Извежда пещерите, чието ниво на трудност е по-малко или равно на
зададената стойност */
find_easy_caves(MaxDifficulty) :-
    cave(Name, Length, Location, Age, Diff),
    Diff =<= MaxDifficulty,
    format('~w: Length = ~w m, Location = ~w, Difficulty = ~w~n',
    [Name, Length, Location, Diff]),    fail.
find_easy_caves(_).

```

Фигура 2. Сегмент от разширената база знания

От учениците са изготвени още правила които извеждат пещерите, които: имат дължина по-голяма или равна на зададената стойност; имат дължина по-малка или равна на зададената стойност; са по-млади или равни на зададената възраст; са по-стари или равни на зададената възраст.

Така поетапно се изгражда базата знания като към нея могат да се добавят и още характеристики (географски координати, надморска височина, температура, животни живеещи в дадените пещери, близки населени места и др.), факти и правила. Например може да се добавят правила за определяне на маршрути, защитени животни и др. Също така към всяка пещера могат да се добавят факти и информация за географско положение и исторически данни.

Реализирането на представения проект е пример (според нас) за елитарно обучение. С този подход се стремим да направим основните концепции на логическото програмиране достъпни за ученици, посредством изучаване и използване на езика за програмиране Пролог. Така с един пример учениците освен да разберат и използват Пролог научават много неща по предметите физика, химия, биология, география, история и т.н. Използва се проектно-базирано обучение в СТЕМ предмет изучаван в училището или в клубове по интереси. Учениците могат да се разделят на групи в зависимост от възможностите им като едните определят каква да е структурата на фактите и правилата, а останалите по направения еталон започват изграждането на базата знания.

Създадената база знания, след одобрение се въвежда в изградена за масово ползване платформа. В тази платформа, екипи или самостоятелно, учители и ученици могат да използват предоставените тематични бази знания за обучение и информация по различни предмети от обучението си в средното училище.

- Помощ за здравословно хранене (подсимволен ИИ)

Дълбочината на теоретичната подготовка е критичен фактор. Докато символният ИИ е достъпен без силен математически апарат, подсимволният ИИ предполага поне базово разбиране на основни концепции от линейната алгебра, теорията на вероятностите и статистиката. Въпреки това, с подходящи инструменти и визуални среди, учениците могат да „проектират“ и използват подсимволни модели без да познават цялата математическа основа [13]. В българските училища учениците от 8. до 12. клас изучават теми, които създават основа за разбиране на подсимволния ИИ, макар тази връзка рядко да се прави явно

в учебните програми¹. В 8. клас се въвежда понятието „вектор“, ключово за представяне на данни в машинното учене. В 10. клас се изучават статистически показатели, а в 11. — условна вероятност и независимост на събития, основополагащи за моделирането. Теми като матрици и производни се разглеждат само в профилирана подготовка по математика. Въпреки това, тези знания често остават откъснати от практическите им приложения. Учениците рядко свързват теорията с въпроси като: „Как векторите описват данни?“, „Как статистиката подпомага моделирането?“, „Защо производната е важна за обучението на модели?“. Без тази връзка мотивацията намалява, а знанията остават абстрактни. Програмирането не е просто техническо умение, а упражнение по логика и последователност — основа на всяко научно мислене. Когато взаимодействието с компютъра се опростява до обикновен език, се губи важният навик за структурирано мислене и причинно-следствени връзки.

Автоматизацията на обучението и възходът на генеративни AI системи може да изглеждат като облекчение, но крият реален риск — да лишат младите хора от най-важното: опита на размишлението, пробите и грешките, постепенното изграждане на логическо мислене. Ако децата не се научат да програмират, те ще бъдат контролирани от онези, които могат. Защото уменията за програмиране не са анахронизъм, а основа на цифровата грамотност.

В търсене на отговор на горните въпроси реализирахме експериментален образователен проект, в който група ученици и преподавател по програмиране работят върху създаването на интелигентна система за анализ на хранителни етикети. Приложението трябва да може да чете текст от етикети, да анализира съставките и да ги класифицира според степента им на риск за здравето. Целта не е непременно да се изгражда изкуствен интелект от нулата — достатъчно е да се използват съществуващи библиотеки, за да се разбере как работят те, как се прилагат на практика и какви математически принципи стоят зад тях. Този подход е не само напълно реалистичен за училищна среда, но и изключително полезен: вместо учениците да останат само пасивни потребители, те се учат да използват налични инструменти и библиотеки, за да решават реални проблеми. По този начин започват да разбират как работи един алгоритъм, какви решения взема и на каква логика се основава. Работейки по конкретна задача — разпознаване и анализ на съставки от хранителни етикети — те прилагат своите знания, учат се да структурират информация, мислят алгоритмично и изграждат нещо с практическа и обществена стойност.

Проектът започва с проучване на хранителни добавки като E250, E621 и E407 — вещества, свързани с повишен риск от онкологични заболявания, алергии, високо кръвно налягане и хормонални нарушения. Въпреки че в интернет съществуват публични списъци и бази данни за добавки и техните ефекти, учениците решават да създадат собствена база данни. Това не само им позволява да съберат информация от надеждни източници, но и да се запознаят по-добре с терминологията, категориите на риск и спецификите на всяка съставка.

Следващата логична стъпка е изборът на технологии. За реализацията учителят запознава учениците с няколко ключови библиотеки в Python: EasyOCR — основният инструмент за разпознаване на текст; OpenCV — за обработка на изображения; Matplotlib — за визуализация на данни; Scikit-learn — за анализ и класификация на съставки; и Pandas — за работа с бази данни и текстова информация. Тъй като Python се изучава на базово ниво в училище, проектът е структуриран с опростен и разбираем код, съобразен с учениците. Вместо да създават собствени модели, те използват готови библиотеки за основни ИИ функции като OCR и класификация на съставки. По инициатива на учителя е използвана библиотеката Streamlit за създаване на уеб приложение. Учениците постепенно усвояват ключови елементи — зареждане на изображения, извличане на текст и проверка в база данни — чрез насочване и готови кодови фрагменти. Не се очаква те самостоятелно да изградят сложна система, особено при първи стъпки в програмирането. Затова ръководната роля на учителя е решаваща — той подбира темата, структурира задачите и подпомага учениците в разбирането на инструментите. Макар да не пишат собствени ИИ алгоритми, учениците започват да разбират как работят библиотеките и концепции като разпознаване на образи, машинно обучение и структуриране на данни. Проектът не цели създаване на нов ИИ, а разбиране как работят съвременните технологии и как математиката, логиката и програмирането намират приложение. Така учениците осъзнават значението на училищни теми като вектори, матрици, производни и статистика — не като абстракции, а като основа на реални ИИ приложения. В началото не разбират напълно какво се случва в библиотеките като EasyOCR или Scikit-learn, но чрез практика и насочване започват да схващат, че изображенията се представят с матрици, че производните се използват в обучението, а статистиката — в моделирането на вероятности.

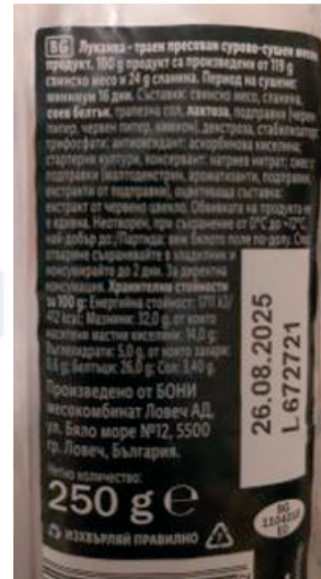
Как работи създаденият прототип? Създаденият прототип може да приеме снимка на етикет, да извлече съставките и да ги класифицира като безопасни, умерено рискови или вредни. Освен това генерира препоръки за по-здравословен избор на храни. Процесът започва с лесна и достъпна стъпка за потребителя — той просто прави или качва снимка на етикета на хранителен продукт (фиг. 3).

Завъртане + OCR на етикет + съставки и алтернативи

Качи изображение на етикет:

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG

Browse files



Фигура 3. Качване на изображение на етикет от хранителен продукт (луканка)

След като снимката е добавена, системата „чете“ текста от изображението и извлича списъка със съставки, който обикновено е отпечатан на етикета. Когато съставките са извлечени, те се сравняват с предварително създадена база данни, която съдържа информация за потенциално опасните, умерено рисковите и безопасни добавки. В случая се използват две библиотеки: EasyOCR и pytesseract, за да разчете текста от снимката – например списък със съставки. Учениците разбират какво представлява оптичното разпознаване на текст (OCR) и виждат как библиотека с вградени ИИ модели може да „прочете“ етикета (фиг. 4).


Разпознат текст:

- BG| Луканка = траен пресован сурово-Суше#| (точност: 0.38)
- продукт 100 g продукт са произведени 0g 1l9 g (точност: 0.33)
- СВИНСКО месо*24g сланиНа: (точност: 0.25)
- на СУШеНе; (точност: 0.54)
- МИНИМУМ 16дни Съставки: СВИНСКО МесоQ; СЛаниНА (точност: 0.25)
- сбеВ белтък трапезна сол лактоза ПОДправК" (точност: 0.21)
- тилпер червен тиер кИЧИОН| декстроза; СтаБИЛМЗаЮ; (точност: 0.09)
- трифоСФатн: анткoкскдант: аскорбиноВа КИСелиНА; (точност: 0.34)
- стартерни клури кнсервант: натриев НТрат; (MGC| (точност: 0.24)
- ПОДправКИ Малоде ср dОМаизалТИ ПОДПОаВН| (точност: 0.05)
- ОКСраКТН ОТЛЮЛьJ)) бвегваца СЪСТЯВКА" (точност: 0.01)
- @СТракт 07Чертеколъекла QбвиVkdТа на ПродуКла Ю (точност: 0.01)
- еЯДИВНА; Неоввтгег =ии сърание 01 О*Сдо I2 ((точност: 0.01)
- Шай (точност: 0.20)
- добърло ППапп ли bot поле ПО ДОЛУ (ic: (точност: 0.02)
- @ВВаРЯНРСьФЛНРИСВЛТОПИИНИК (точност: 0.04)


Фигура 4. Разпознаване на текста от етикета


Освен това, се използва библиотеката OpenCV, за да се обработи изображението – например да се завърти автоматично, ако е снимано накриво. Това показва как ИИ може да се комбинира с компютърно зрение. След това приложението проверява дали в разпознатия текст се срещат вредни Е-та или думи като „глутамат“, „нитрит“, „карагинан“. Резултатите се визуализират чрез различни индикатори, което не само улеснява потребителя в бързото разбиране на информацията, но и запознава учениците с концепцията за категоризация на данни и изграждане на интуитивен потребителски интерфейс (фиг. 5).

 **Вредни съставки (Е-номера):**

 Няма открити вредни Е-номера.

 **Засечени съставки (по дума):**


 фосфат – Фосфати – могат да влияят негативно на бъбреците


 лактоза – Лактоза – може да причини стомашен дискомфорт при непоносимост


Фигура 5. Резултат от действието на системата

Ако продуктът съдържа вредни съставки, системата предлага здравословни алтернативи, базирани на типа храна (напр. „преработено месо“) (фиг. 6). Учениците изграждат логика за категоризация, която отново използва речници и сравняване на думи – умения, които са част от основната работа с текст в Python.

 **Алтернативи на Преработено месо:**

 Вместо колбас – печено пилешко филе с подправки.

 Леща яхния с моркови и подправки.

 Яйца с авокадо и свежи зеленчуци.

Фигура 6. Предложени здравословни алтернативи на типа храна

Учениците в процеса на работа се учат да интегрират технологии, свързани с компютърно зрение, обработка на естествен език, бази данни и уеб програмиране.

Заключение. С нарастващата достъпност на ИИ инструментите все повече ученици проявяват интерес към използването им. Създаването на собствени ИИ модели обаче е по-сложен въпрос, особено за ученици в начална и гимназиална степен. Нашият опит показва, че логическото програмиране (напр. с Пролог) е подходящо за начално ниво, особено в области, изискващи прозрачност и разсъждение.

Проектите с подсимволен ИИ (напр. невронни мрежи) показват, че училищните математически знания са полезна основа, но недостатъчни за пълно разбиране без допълнителна подготовка. Учениците могат да използват готови инструменти, но не и самостоятелно да създават модели или алгоритми на дълбоко ниво. Затова ИИ обучението в училище трябва да бъде адаптивно – да осигурява достъпна технологична грамотност за всички, но и възможности за напреднали ученици. Подходящото учителско насочване и специализирани обучения могат да позволят на мотивирани ученици да задълбочат познанията си в ИИ.

Необходим е баланс между достъпност и дълбочина, между масово и индивидуализирано обучение. Чрез диференцирано изучаване на символни и подсимволни методи и връзка с други дисциплини, учениците могат да развият критично мислене и практически умения. Те могат да създават ИИ, макар и в ограничени рамки, ако обучението е поэтапно и подкрепено от учители и експерти. Целта е не просто използване, а съзнателно и отговорно създаване на интелигентни системи

References:

1. Nils J. Nilsson. 1998. Artificial Intelligence: A New Synthesis. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN:978-0-08-049945-1, pp.513.
2. Goodfellow, I., et al. (2016) Deep Learning. MIT Press, Cambridge, MA. <http://www.deeplearningbook.org>
3. J. C. Maxwell, The Scienti_c Letters and Papers of James Clerk Maxwell: 1846-1862, Cambridge: Cambridge University Press, 1990.
4. M. Born, The Born Einstein Letters, Walker and Company, 1971.
5. K. Gold, Norvig vs. Chomsky and the Fight for the Future of AI, TOR.COM, 21 June 2011.

6. Maruyama, Y. (2021). Symbolic and Statistical Theories of Cognition: Towards Integrated Artificial Intelligence. In: Cleophas, L., Massink, M. (eds) Software Engineering and Formal Methods. SEFM 2020 Collocated Workshops. SEFM 2020. Lecture Notes in Computer Science(), vol 12524. Springer, Cham.
7. K. Bansal, S. M. Loos, M. N. Rabe, C. Szegedy, and S. Wilcox, HOList: An Environment for Machine Learning of Higher Order Logic Theorem Proving, Proceedings of ICML, pp. 454-463, 2019.
8. Besold et al., Neural-Symbolic Learning and Reasoning: A Survey and Interpretation, 2017, Artificial Intelligence (cs.AI), arXiv:1711.03902, <https://doi.org/10.48550/arXiv.1711.03902>.
9. M. Frixione, G. Spinelli, S. Gaglio, Symbols and subsymbols for representing knowledge: a catalogue raisonne, in: Proceedings of the 11th international joint conference on Artificial Intelligence-Volume 1, 1989, pp. 3–7.
10. M. L. Minsky, Logical Versus Analogical or Symbolic Versus Connectionist or Neat Versus Scru_y, AI Magazine, vol. 12, pp. 34-51, 1991.
11. N. C. Thompson et al., The Computational Limits of Deep Learning, arXiv:2007.05558, 2020.
12. Cecchi, LA, Rodríguez, JP, Dahl, V. (2023). Logic Programming at Elementary School: Why, What and How Should We Teach Logic Programming to Children?. In: Warren, DS, Dahl, V., Eiter, T., Hermenegildo, MV, Kowalski, R., Rossi, F. (eds) Prolog: The Next 50 Years. Lecture Notes in Computer Science(), vol 13900. Springer, Cham. https://doi.org/10.1007/978-3-031-35254-6_11.

(Endnotes:)

1. https://www.mon.bg/obshto-obrazovanie/uchebni-planove-i-programi-2/uchebni-programi/uchebni-programi-za-obsthoobrazovatelna-podgotovka/Uчебni_programi_po_matematika