https://doi.org/10.53656/math2026-5-3-pan

Science in Education
Научно-методически статии

П.А.Н. В КЛАСНАТА СТАЯ: ПРАКТИЧЕН ПОДХОД ЗА ПО-УСПЕШНО ОБУЧЕНИЕ ПО ПРОГРАМИРАНЕ НА PYTHON В 8. КЛАС

Илия Добрев

Професионална гимназия по архитектура, строителство и геодезия „Арх. Камен Петков“, Пловдив (България)

Резюме. Статията разглежда предизвикателствата при преподаването на текстов програмен език (Python) в началото на гимназиалния етап, когато учениците постъпват с неравномерна предварителна подготовка и различна увереност при писане на програмен код. Фокусът е върху преодоляването на дисбаланса чрез трансфер на принципи от чуждоезиковото обучение и прилагане на методическата рамка П.А.Н. (планиране, активно учене, непрекъснати упражнения) в проектно базирана среда. Предлага се педагогически сценарий за работа със структурите от данни „списъци“ и „речници“ в контекста на ученически проект „Хранителен калкулатор“. Чрез последователно „езиково“ въвеждане на понятия (лексика, синтаксис, семантика на кода), контролирани практики и спираловидно упражняване се демонстрира как преходът към асоциативни структури улеснява моделирането на реални данни, повишава четливостта и подпомага формирането на устойчиви навици в програмирането. Обсъждат се индикатори за напредък: точност на решенията, качество на кода и самооценка на компетентностите.

Ключови думи: програмиране на Python; проектно базирано обучение; метод П.А.Н.; изравняване на компетентности; структури от данни

1. Въведение

В 8. клас обучението по програмиране на Python често започва в условия на силно разнородни входни нива – част от учениците имат опит от клубове, платформи и състезателни среди, докато други тепърва изграждат базова компютърна грамотност и логическо моделиране. Тази асиметрия създава две трудности: напредналите ученици губят мотивация при прекомерно темпо „за начинаещи“, а учениците без опит бързо натрупват дефицити и развиват избягващо поведение („това не е за мен“).

В настоящата разработка „изравняване на компетентностите“ не се разбира като уеднаквяване на темпото за всички ученици, а като създаване на учебна среда, в която всеки ученик има достъп до минимално необходимите опори за успешно изпълнение на задачата. Подходът П.А.Н. реализира това чрез три механизма: първо, планиране на малки и проверими стъпки; второ, активно учене чрез практически проблем и работа по роли; трето, непрекъснати упражнения, които позволяват многократно връщане към основните конструкции — списъци, речници, достъп до елемент, търсене и проверка на резултат. По този начин учениците с по-ниско входно ниво получават структурирана подкрепа, а учениците с по-висока подготовка могат да работят върху разширения на проекта.

Актуалността на проблема се усилва от факта, че Python се въвежда като текстов език, при който дори малки синтактични грешки могат да доведат до неработеща програма. Ранното преминаване към структури от данни като списъци и речници допълнително увеличава когнитивното натоварване, тъй като учениците трябва едновременно да овладеят синтаксис, логика на изпълнение и модел на представяне на данните.

Настоящата разработка предлага елемент на новост: трансфер на дидактически принципи от чуждоезиковото обучение към обучението по програмиране чрез рамката П.А.Н. (планиране, активно учене, непрекъснати упражнения) в проектно базирана среда. Рамката П.А.Н. осигурява „мостове“ за учениците с дефицити, като същевременно създава условия за развитието на дигитални компетентности в средното училище (Redecker & Punie, 2017).

2. Изложение

Теоретичен обзор: „лингвистична“ перспектива към кода. Кодът може да се разглежда като формален език, чрез който ученикът овладява:

- лексика (ключови думи, идентификатори, типове данни);
- синтаксис (правила за запис – скоби, двоеточия, отстъпи, структури);
- семантика (смисъл – какво реално се случва при изпълнение).

Усвояването на тези компоненти се постига най-ефективно чрез намаляване на когнитивните бариери и използване на структурирана система от задачи (Kelleher & Pausch, 2005)

При начинаещите в програмирането честа причина за грешки е смесването на тези нива. Ученикът „познава“ дума или конструкция, но не я използва смислено в контекст или разбира смисъла, но не владее правилния запис. Това наподобява типични трудности при изучаване на чужд език (знам думата, но не я употребявам правилно в изречение). Ефективното преодоляване на тези трудности изисква специфичен подход към управляването на грешките и развиване на фундаментално алгоритмично мислене (Wing, 2006).

Оттук следва, че методическите решения могат да бъдат структурирани подобно на езиково обучение – въвеждане → контролирана практика → комуникативна практика. В програмирането би било – минизадачи → насочени упражнения → проект.

3. Методическа рамка П.А.Н. в проектно базирано обучение по Python

Рамката П.А.Н. се разглежда като цикъл от три взаимосвързани компонента: планиране, активно учене и непрекъснати упражнения.

Планирането включва няколко етапа, а именно:

- ясно дефинирани цели за урок/етап (какво „може“ ученикът да направи с кода);
- микростъпки и скеле (шаблони, частично попълнен код, тестови данни);
- предвиждане на типични грешки и контрамерки (напр. “KeyError”, “IndexError”, грешно форматиране на вход).

При активното учене ученикът работи: с реален проблем и взема решения (структура на данните, функции, вход/изход); по роли (анализатор данни/програмист/тестер/документатор); с кратки рефлексивни паузи: „Какво избрах и защо?“.

Третият компонент от методическата рамка П.А.Н. са непрекъснатите упражнения. Те се състоят от спираловидно връщане към едни и същи концепти в нов контекст; микрозадачи за

автоматизация (5 – 8 мин.), минитестове, „код диктовка“ (възпроизвеждане на кратък фрагмент по модел); кратки рубрики за качество (четимост, именуване, структурираност). Значението на системната практика и непосредствената обратна връзка се откроява и в друг предметен контекст. Тодоров установява, че игрово организираните дигитални упражнения с незабавна обратна връзка подпомагат автоматизирането на уменията и същевременно намаляват емоционалното напрежение, свързано с допускането на грешки (Todorov, 2026).

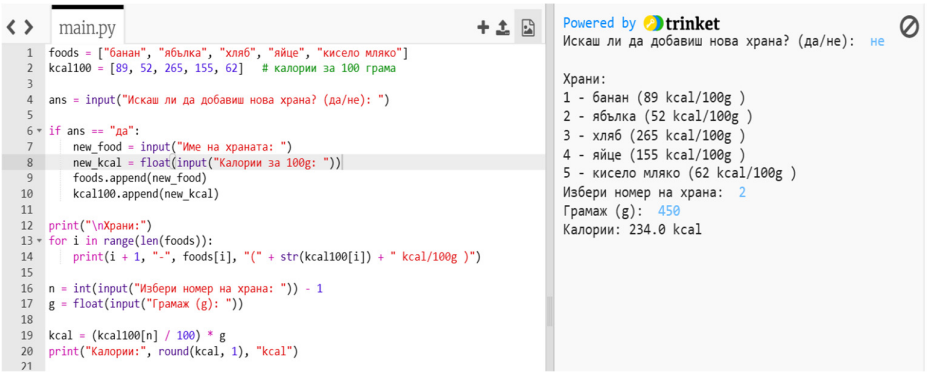
Подходът се основава на разбирането, че логическата последователност, абстракцията и алгоритмичното мислене се развиват най-ефективно чрез активно учене, проблемно ориентирани задачи и прилагане на знанието в реален контекст (Shopova & Radev, 2026). Практически изследвания върху прилагането на STEM подхода в училищна среда също показват, че интерактивните технологии, работата по практически задачи и съвместното планиране на дейностите създават условия за развитие на критическо мислене, екипна работа и приложение на знанията в реални ситуации (Shopova & Dimitrov, 2025). В този смисъл проектният сценарий „Хранителен калкулатор“ е разгледан не само като упражнение по синтаксис, а като среда за моделиране на данни, проверка на решения и аргументиран избор на структура.

Дидактически сценарий „Хранителен калкулатор“ като контекст за списъци (фиг. 1) и речници (фиг. 2).

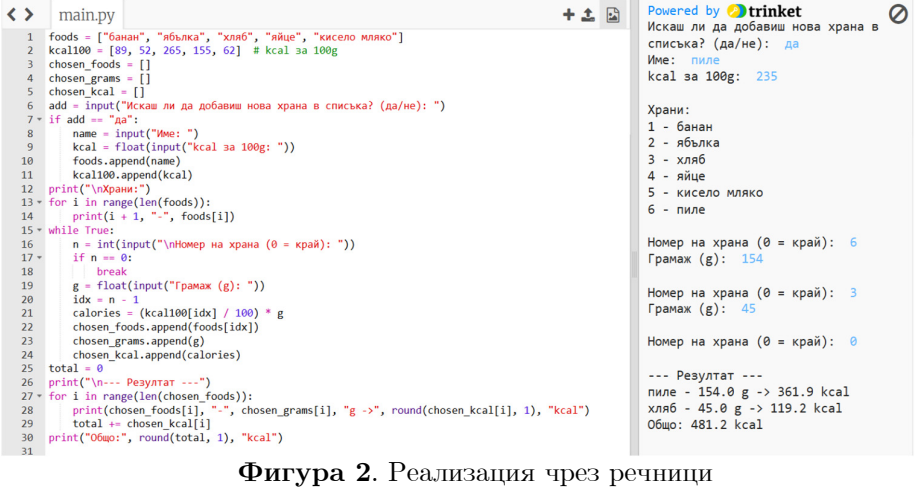
Проектът „Хранителен калкулатор“ е подходящ, защото работи с реални данни (продукти, калории, грамажи) и естествено изисква структуриране. Списъците позволяват последователност, но при по-сложни операции (търсене по име, извличане на стойност) учениците често стигат до нечетими решения с паралелни списъци или сложни индекси.

Преходът към речници (асоциативни масиви) въвежда по-интуитивен модел: „име → стойност“, което съвпада с начина, по който човек мисли за справочник. Това намалява „шума“ в кода и освобождава капацитет за алгоритмично мислене и проверка на

резултатите (фиг. 2). Използването на структури от данни за моделиране на хранителни стойности позволява интеграция на реални данни в обучението по Python и създава условия за достъпно и ангажиращо програмиране за всички ученици (Resnick et al., 2009).



Фигура 1. Реализация чрез списък



Фигура 2. Реализация чрез речници

3.1. Методология на проекта

Проектно базираното обучение позволява компетентностите да се формират чрез система от взаимосвързани практически задачи, като предполага етапност, конкретни резултати и активно участие на

учениците (Atanasova, 2023). Проектно базираното обучение е с продължителност от един месец, като се прилага в задължителните часове на ученици от осми клас по предмета информационни технологии в ПГАСГ „Арх. Камен Петков“, Пловдив.

Дизайн на изследването. Изследването е квазиекспериментално (нерандомизирано), с експериментална група (n=107) и контролна група (n=52). Групите са формирани по паралелки. За проверка на базова сравнимост е приложен тест върху списъци и речници (15 мин.), като се анализират средни резултати и разсейване по групи. Интервенцията се провежда в рамките на 4 седмици по един час седмично в задължителните часове по ИТ.

Обект на изследване е процесът на овладяване на структури от данни в начално обучение по програмиране на Python в 8. клас.

Предмет на изследване е ефектът от прилагането на П.А.Н. в проектно базирано обучение върху: разбиране и употреба на списъци и речници; четимост и структурираност на кода; увереност при програмиране.

Целта на обучението е да се провери дали внедряването на П.А.Н. в рамките на проект „Хранителен калкулатор“ подпомага изравняването на компетентностите и подобрява качеството на програмните решения при ученици в 8. клас.

Задачите, които се поставят с този експеримент, са: да се разработи педагогически сценарий за проекта „Хранителен калкулатор“, включващ преход от списъци към речници; да се приложи интервенция по модела П.А.Н. в рамките на проектно базирани занятия; да се сравнят резултатите преди и след интервенцията по показатели за точност, четимост и увереност.

Хипотезата на проекта: очаква се учениците, работещи по П.А.Н., да покажат по-високи резултати при решаване на задачите с речници спрямо учениците, които работят без системна рамка.

3.2. Приложение на метода П.А.Н. в проекта „Хранителен калкулатор“

Цел: Преодоляване на „синдрома на белия лист“ и формулиране на логиката на естествен език.

Сценарий 3	
Учителят (Ментор в сянка)	Отстъпва назад. Учителят спира да дава инструкции стъпка-по-стъпка. Той задава отворени предизвикателства: *„А какво ще стане, ако добавим и напитки?“*. Ролята му е само да подкрепя и да валидира успехите.
Ученикът (Създател)	Експериментира. Тъй като синтаксисът е лесен, ученикът започва да си играе с кода. Той добавя свои любими храни, променя калориите, тества какво става, ако потърси несъществуващ продукт. Страхът от грешка е заменен от любопитство.
Резултат	Преход от пасивно възпроизвеждане към активно творчество.

След провеждане на проектно базираното обучение се наблюдават: увеличение на средния резултат на задачите с речници; спад на честотата на синтактични грешки, свързани с достъп до елементи (индексиране/ключове), след като учениците започнат да мислят за „име → стойност“ вместо за позиция; по-високи оценки по рубриката за четимост, най-вече при критериите „смислено именуване“ и „структурирани данни“.

3.3. Резултати от приложението

Инструментите за отчитане на резултатите от проекта са: (1) тестове преди и след обучението (10 – 15 минутни), (2) самооценъчна скала (1–5) за увереност при четене на код, писане на функции и работа с данни, и (3) дневник на наблюденията (типични грешки, реакция на учениците, време за изпълнение).

Преди старта на проектно базираното обучение по модела П.А.Н. и след протичането му учениците решават тест за списъци и речници в рамките на 15 минути. Учителят отбелязва типични грешки, реакции и време за изпълнение по време на проекта, а в края учениците попълват самооценка на увереността.

За измерване на напредъка е използван кратък диагностичен тест, проведен преди и след интервенцията. Тестът е с продължителност 15 минути и включва 3 задачи с общ максимален резултат 10 точки. Първата задача проверява разчитане на кратък фрагмент код със списък или речник – 3 точки. Втората задача изисква добавяне или извличане на елемент по индекс/ключ – 3 точки. Третата задача

Сценарий 3	
Учителят (Ментор в сянка)	Отстъпва назад. Учителят спира да дава инструкции стъпка-по-стъпка. Той задава отворени предизвикателства: *„А какво ще стане, ако добавим и напитки?“*. Ролята му е само да подкрепя и да валидира успехите.
Ученикът (Създател)	Експериментира. Тъй като синтаксисът е лесен, ученикът започва да си играе с кода. Той добавя свои любими храни, променя калориите, тества какво става, ако потърси несъществуващ продукт. Страхът от грешка е заменен от любопитство.
Резултат	Преход от пасивно възпроизвеждане към активно творчество.

След провеждане на проектно базираното обучение се наблюдават: увеличение на средния резултат на задачите с речници; спад на честотата на синтактични грешки, свързани с достъп до елементи (индексиране/ключове), след като учениците започнат да мислят за „име → стойност“ вместо за позиция; по-високи оценки по рубриката за четимост, най-вече при критериите „смислено именуване“ и „структурирани данни“.

3.3. Резултати от приложението

Инструментите за отчитане на резултатите от проекта са: (1) тестове преди и след обучението (10 – 15 минутни), (2) самооценъчна скала (1–5) за увереност при четене на код, писане на функции и работа с данни, и (3) дневник на наблюденията (типични грешки, реакция на учениците, време за изпълнение).

Преди старта на проектно базираното обучение по модела П.А.Н. и след протичането му учениците решават тест за списъци и речници в рамките на 15 минути. Учителят отбелязва типични грешки, реакции и време за изпълнение по време на проекта, а в края учениците попълват самооценка на увереността.

За измерване на напредъка е използван кратък диагностичен тест, проведен преди и след интервенцията. Тестът е с продължителност 15 минути и включва 3 задачи с общ максимален резултат 10 точки. Първата задача проверява разчитане на кратък фрагмент код със списък или речник – 3 точки. Втората задача изисква добавяне или извличане на елемент по индекс/ключ – 3 точки. Третата задача

изисква реализиране на елементарна функция за търсене или изчисление върху зададени данни – 4 точки.

Входящият и изходящият тест са еквивалентни по структура, трудност и критерии за оценяване, но използват различни конкретни данни, за да се ограничи ефектът от запомняне на отговори. Например във входящия тест се работи с продукти и калории, а в изходящия — с предмети и цени/точки, като проверяваните операции остават еднакви.

В експерименталната група делът на учениците, които не се справят с теста, намалява от 78.5% (84/107) преди интервенцията до 21.5% (23/107) след нея, което представлява спад от 57.0 процентни пункта. В контролната група спадът е от 90.4% (47/52) до 61.5% (32/52), т.е. 28.8 процентни пункта. Сравнението на промените между групите показва по-изразено намаляване на неуспеха в експерименталната група (разликата е ≈ 28.2 процентни пункта в полза на ЕГ). При изходящото измерване разликата между групите е статистически значима ($\chi^2(1)=24.79$, $p<.001$), докато при входящото измерване не достига конвенционалното ниво на значимост ($\chi^2(1)=3.41$, $p\approx.065$). Резултатите са показани в табл. 2.

Таблица 2. Сравнителен анализ (преди/след)

Група	**n**	**Преди: неуспех, n (%)**	**След: неуспех, n (%)**	**Промяна (pp)**
Експериментална група	107	84 (78.5%)	23 (21.5%)	−57.0
Контролна група	52	47 (90.4%)	32 (61.5%)	−28.8

Забележка. „Неуспех“ е дефиниран като неизпълнение на поне 2 от 3 ключови операции: (а) разчитане на код със списък/речник, (б) добавяне/извличане на елемент по ключ, (в) реализиране на елементарна функция за търсене. Промяната е изчислена в процентни пункта (pp) и е представена като дял ученици, класифицирани като „не се справя“, преди и след интервенцията.

Поради различната численост на експерименталната и контролната група резултатите се представят не само чрез абсолютен брой ученици, но и чрез относителни дялове в проценти. За проверка на базовата съпоставимост е сравнено входящото ниво на двете групи чрез χ^2-критерий за независимост. При входящото измерване разликата между групите не достига конвенционалното ниво на статистическа значимост, но показва тенденция към по-слабо начално представяне на контролната група. Поради това резултатите се интерпретират предпазливо и се разглеждат като индикативни, а не като доказателство за категорична причинно-следствена връзка.

Основният показател за сравнение е промяната в дела на учениците, които не се справят с теста преди и след обучението. Това позволява да се проследи не само крайният резултат, но и степента на подобрение във всяка група.

3.4. Обсъждане

Намаляването на дела на неуспехите е по-изразено в експерименталната група, което е съвместимо с хипотезата, че рамката П.А.Н. подпомага овладяването чрез структурирано скеле, контролирана практика и спираловидно упражняване в проектен контекст. Въпреки това резултатите следва да се интерпретират предпазливо поради квазиекспериментални (нерандомизиран) дизайн и възможни начални различия между паралелките; затова по-подходящо е да се говори за индикативен ефект, а не за категорична причинност.

Като следваща стъпка се препоръчва стандартизиране на прага за „успех“, по-дълго проследяване (поне един срок) и проверка на трансфер към други теми (напр. файлове, API, ООП). В бъдеще методът може да се оптимизира чрез прилагането на интерактивни дигитални платформи за оценка и развиване на логическото мислене (Thomas, 2000; Shopova & Radev, 2026).

3.5. Практически импликации

На база наблюдаваното намаляване дела на неуспехите и качествените наблюдения по време на проекта могат да се формулират

следните практически насоки за преподаването в началните модули по Python:

- Използване на П.А.Н. като рамка за планиране и провеждане на занятията (ясни микростъпки, контролирана практика и циклично упражняване).
- Дизайн на задачи с целенасочен „концептуален скок“ от списък към речник, при който учениците преживяват ограниченията на индексното мислене и аргументирано преминават към асоциативен модел.
- Въвеждане на рубрики за качество на кода още в 8. клас (четимост, именуване, структуриране на данните), така че оценяването да не се свежда единствено до „работи/не работи“.

  Критериите за оценяване са представени в табл. 3.

Таблица 3. Критерии на оценяване

Критерий	**0 т.**	**1 т.**	**2 т.**
Смислено именуване	Имената са неясни или произволни	Част от имената са подходящи	Имената ясно описват предназначението
Структуриране на данните	Данните са разпилени или неподходящо представени	Използвана е структура, но с частични неточности	Използвана е подходяща структура: списък/речник
Четимост на кода	Кодът е труден за проследяване	Кодът е частично подреден	Кодът е логически подреден и лесен за проследяване
Проверка на резултата	Липсва проверка или обработка на грешки	Има частична проверка	Има адекватна проверка или реакция при липсващи данни

Максималният резултат по рубриката е 8 точки. За ограничаване на субективността оценяването се извършва по предварително зададени дескриптори. При спорни случаи се използва повторна проверка на кода по същата рубрика, като се разглеждат само наблюдаеми

характеристики: имена на променливи, избор на структура от данни, логическа подредба и наличие на проверка при търсене.

3.6. Ограничения и насоки за бъдещи изследвания

Настоящото изследване е квазиекспериментално и се провежда в рамките на един месец, което ограничава обобщеността на резултатите и не позволява категорични причинно-следствени заключения. В следващи разработки е уместно:

- по-дългосрочно проследяване (поне един учебен срок) и проверка на трансфер към други теми (напр. файлове, API, ООП);
- сравнение на различни проектни контексти (финансов калкулатор, дневник на оценки, инвентарна система), за да се провери доколко ефектът зависи от сюжета и близостта до реални данни;
- анализ на типове грешки (напр. KeyError/IndexError, грешно именуване, липса на проверки) и динамиката им по време на цикъла П.А.Н., включително кои подпори работят за различни подгрупи ученици.

4. Заключение

Резултатите са индикативни, тъй като интервенцията е с продължителност един месец и групите са нерандомизирани. Данните от преди/след измерването показват различно входящо ниво между групите и последващо сближаване на средните стойности, като при експерименталната група се наблюдава значително нарастване на вариативността на резултатите. Това насочва към извод, че при част от учениците подходът подпомага овладяването и трансфера, но при друга част са необходими допълнителни подкрепящи стъпки (по-малки задания, повече упражнения, насочена обратна връзка). Необходимо е проследяване в рамките на срок и проверка за трансфер към други теми. Наличните учебни часове не са достатъчни за формиране на трайни стратегии на мислене и за устойчиво самостоятелно вземане на решения.

ЛИТЕРАТУРА

Атанасова, Н. (2023). Реализиране на компетентностния подход чрез учебния проект в началните класове. *Педагогика*, *95*(7), 917 – 927. https://doi.org/10.53656/ped2023-7.5

Велчева, И., Гаров, К. (2021). Дигитални инструменти за изграждане на уеб съдържание в обучението на учители. *Proceedings of REMIA'2021*, 67 – 74. ISBN: 978-619-202-714

Келехър, К., Пауш, Р. (2005). Преодоляване на бариерите в програмирането: Таксономия на среди и езици за начинаещи програмисти. *ACM Computing Surveys*, *37*(2), 83 – 137. https://doi.org/10.1145/1089733.1089734

Редекер, К., & Пуние, И. (2017). *Европейска рамка за дигитална компетентност на преподавателите: DigCompEdu*. Publications Office of the European Union. https://doi.org/10.2760/159770

Резник, М., и др. (2009). Scratch: Програмиране за всички. *Communications of the ACM*, *52*(11), 60 – 67. https://doi.org/10.1145/1592761.1592779

Тодоров, М. (2026). Геймификация на обучението по правопис в началното образование чрез платформата Educandy. *The 3rd EIID International Conference "Pedagogy and Psychology of Trust and Learner Agency in the Age of Generative Systems"*, ESEJ, 86 – 95. https://doi.org/10.47451/esej-ped-41

Томас, Дж. У. (2000). *Преглед на изследванията върху проектно-базираното обучение*. The Autodesk Foundation.

Уинг, Дж. М. (2006). Компютърно мислене. *Communications of the ACM*, *49*(3), 33 – 35. https://doi.org/10.1145/1118178.1118215

Шопова, В., & Димитров, И. (2025). Изкуствен интелект в подкрепа на STEM обучението в класната стая. *International Conference on Virtual Learning*, (20), 93 – 102. https://doi.org/10.58503/icvl-v20y202508

Шопова, В., & Радев, В. (2026). Интерактивни платформи за развиване на математическо мислене в пети клас. *International Conference on Virtual Learning*, (21), 139 – 145. https://doi.org/10.58503/icvl-v21y202612

REFERENCES

Atanasova, N. (2023). Implementation of the competence approach through the learning project in primary grades. *Pedagogy*, *95*(7), 917 – 927. DOI: 10.53656/ped2023-7.5

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys*, *37*(2), 83 – 137. DOI: 10.1145/1089733.1089734

Redecker, C., & Punie, Y. (2017). *European Framework for the Digital Competence of Educators: DigCompEdu*. Publications Office of the European Union. DOI: 10.2760/159770

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM*, *52*(11), 60 – 67. DOI: 10.1145/1592761.1592779

Shopova, V., & Dimitrov, I. (2025). Artificial Intelligence to support STEM learning in the classroom. *International Conference on Virtual Learning*, (20), 93 – 102. DOI: 10.58503/icvl-v20y202508

Shopova, V., & Radev, V. (2026). Interactive platforms for developing mathematical thinking in fifth grade. *International Conference on Virtual Learning*, (21), 139 – 145. DOI: 10.58503/icvl-v21y202612

Thomas, J. W. (2000). *A review of research on project-based learning*. The Autodesk Foundation.

Todorov, M. (2026). Gamification of spelling instruction in primary education through the Educandy platform. *The 3rd EIID International Conference "Pedagogy and Psychology of Trust and Learner Agency in the Age of Generative Systems"*, ESEJ, 86 – 95. DOI: 10.47451/esej-ped-41

Velcheva, I., & Garov, K. (2021). Digital Tools for Building Web Content in Teachers' Training. *Proceedings: Intelligent innovative ICT in research in mathematics, informatics, and pedagogy of education (REMIA'2021)*, 67 – 74. ISBN: 978-619-202-714

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33 – 35. DOI: 10.1145/1118178.1118215

P.A.N. IN THE CLASSROOM: A PRACTICAL APPROACH TO MORE SUCCESSFUL PYTHON PROGRAMMING EDUCATION IN 8th GRADE

Abstract. This article examines the challenges of teaching a text-based programming language (Python) at the beginning of the secondary school stage, when students enter with uneven prior preparation and varying levels of confidence in writing program code. The focus is on overcoming this imbalance by transferring principles from foreign language teaching and applying the P.A.N. (Planning, Active Learning, Non-stop Practice) methodological framework within a project-based environment. A pedagogical scenario is proposed for working with the list and dictionary data structures in the context of a student project titled "Nutrition Calculator". Through a sequential "linguistic" introduction of concepts (lexicon, syntax, and semantics of code), controlled practices, and spiral learning exercises, the article demonstrates how the transition to associative structures facilitates the modeling of real-world data, enhances code readability, and supports the formation of sustainable programming habits. Indicators of progress are discussed, including solution accuracy, code quality, and self-assessment of competencies.

Keywords: Python programming; project-based learning; P.A.N. method; leveling of competencies; data structures

✉ **Iliya Dobrev**
Architecture, Construction and Geodesy High School "Arch. Kamen Petkov"
Plovdiv, Bulgaria
E-mail: dobrev.i@pgasg-plovdiv.com