

## КОНЦЕПЦІЇ ПРОЕКТУВАННЯ СИСТЕМ КОНТРОЛЮ ЗНАНЬ З МАТЕМАТИЧНИХ ДИСЦИПЛІН В УМОВАХ ДИСТАНЦІЙНОЇ ОСВІТИ

*Анотація. Пропонується новий підхід до проектування систем контролю знань здобувачів освіти з дисципліни «Диференціальне числення». На відміну від більшості існуючих систем контролю знань авторська розробка не містить баз завдань та відповідей. Завдання генеруються у відповідності до сформульованих правил з використанням випадкових чисел. Згенерована задача (функція, для якої треба знайти похідну) перевіряється на коректність, після чого пропонується здобувачу. Відповідь на завдання надається у вигляді формули, а не числа, що також є відмінною рисою авторської розробки. Висновок про вірний розв'язок відбувається на підставі порівняння результатів аналітичного та чисельного диференціювання заданої функції для декількох значень аргументу*

*Ключові слова: дистанційне навчання, система контролю знань, диференціальне числення, система правил для генерації завдань, алгоритм оцінки відповідей.*

**Постановка проблеми.** У роботі пропонується новий підхід до проектування систем контролю знань для здобувачів освіти з дисципліни «Диференціальне числення», що враховує сучасні виклики дистанційного навчання та потреби у персоналізованому підході до навчання. Традиційні системи контролю знань зазвичай використовують статичні бази завдань і відповідей, що обмежує їхню гнучкість та можливість індивідуального підходу до кожного студента. Авторська розробка кардинально відрізняється від таких систем завдяки динамічному підходу до генерації завдань. Завдання формуються автоматично на основі заданих правил з використанням випадкових чисел, що дозволяє забезпечити унікальність кожного завдання та адаптацію під індивідуальні можливості кожного здобувача освіти.

Згенеровані завдання являють собою функції, для яких необхідно знайти похідну. Важливим етапом є перевірка коректності згенерованого завдання перед його поданням здобувачу освіти. Це забезпечує уникнення помилок у формулюванні завдань та підвищує точність контролю. Особливістю запропонованого підходу є форма надання відповіді: вона подається не у вигляді числового значення, а у вигляді математичної формули, що дозволяє оцінювати не тільки правильність розв'язку, але й глибину розуміння здобувачем методів диференціювання та математичних принципів.

**Аналіз останніх досліджень та публікацій.** Розвиток технологій у сфері обробки символічних виразів, чисельних методів обчислення та машинного навчання відкриває

нові можливості для створення інтелектуальних систем, які здатні ефективно оцінювати та контролювати знання студентів щодо вивчених математичних концепцій. Такі системи можуть автоматично адаптуватися до потреб користувача, надаючи індивідуалізований підхід до контролю якості знань та надаючи зворотний зв'язок для подальшого вдосконалення навчання.

Враховуючи велику кількість здобувачів освіти у сучасних освітніх установах та потребу в ефективному контролі їхніх знань з математики, розробка інтелектуальних інформаційних систем для цієї цілі може мати значний практичний вплив на якість освіти та підготовку фахівців у цій області. Такі системи можуть сприяти покращенню результатів навчання та забезпечити більш об'єктивну оцінку знань студентів. Вони доступні для виконання в мережі та не вимагають реєстрації користувача, що робить їх зручними для використання у навчальних процесах.

Одним з таких напрямків є платформи для створення тестів. Ці платформи надають інструменти для створення різноманітних тестів з математики, які зазвичай використовують базу задач, базуються на стандартизованих або заздалегідь підготовлених наборах завдань [1]. Такі системи мають низку характеристик та архітектурних підходів, що визначають їх функціональні можливості.

Тестові системи з фіксованими завданнями являють собою найбільш поширений тип систем, де користувачам пропонуються завдання з бази даних, які підготовлені викладачами або розробниками курсу. Особливостями таких систем є:

- наявність заздалегідь складених тестів з варіантами відповідей або відкритими питаннями;
- здійснення оцінювання на порівнянні відповіді користувача з коректною відповіддю в базі даних;
- використання, як правило, для перевірки базових знань або швидкого оцінювання.

Прикладом таких систем є тести на освітніх платформах, таких як Moodle [2], де викладачі можуть створювати питання на тему "Диференціальне числення" з варіантами відповідей або формулюванням відкритих питань.

Інтерактивні тренажери включають завдання, що спрямовані на інтерактивне навчання з можливістю перевірки відповідей в реальному часі. Особливостями таких тренажерів є:

- завдання вибираються з фіксованого набору, проте можуть мати певні варіації (наприклад, різні параметри у функціях, для яких потрібно знайти розв'язок);
- підказки та покроковий розбір рішень для самонавчання;
- можливість інтеграції з іншими навчальними матеріалами (відеоуроки, теоретичні довідники).

Прикладом таких систем можна назвати Khan Academy [3], де користувач може отримати завдання на знаходження похідних і після кожного кроку отримати підказки або пояснення.

Системи з адаптивним навчанням. Це системи, що використовують базу завдань і алгоритми адаптивного навчання для автоматичного підбору завдань залежно від рівня підготовки учня. Особливості цих систем є:

- база завдань зазвичай має структуру з завданнями різної складності;
- система аналізує результати попередніх спроб і адаптує наступні завдання для більш точного оцінювання або для вдосконалення знань;
- можуть містити як теоретичні, так і практичні завдання.

Прикладом такої системи може виступати ALEKS (Assessment and Learning in Knowledge Spaces) [4] – система, що адаптує навчальний план залежно від відповідей учня і пропонує нові завдання для зміцнення слабких знань.

Онлайн-платформи для екзаменів та контрольних робіт - це системи використовуються для проведення тестів, контрольних та екзаменів у режимі реального часу. Викладачі завантажують завдання у базу даних, а студенти виконують їх за певний час. Особливості цієї системи такі:

- можливість автоматичного оцінювання завдань (наприклад, якщо відповідь є числовою або функціональною);
- підтримка різних типів питань: відкриті питання, завдання з вибором відповіді, тощо;
- інтеграція з системами дистанційного навчання для відстеження прогресу студента.

Прикладом таких платформ є Proctorio [5] та ExamSoft [6], які використовуються для проведення онлайн-екзаменів.

На основі наведених інформаційних систем з базами задач можна визначити такі ключові особливості:

- фіксованість завдань: усі завдання зберігаються в базі даних і надаються користувачам у певній послідовності або випадковому порядку;
- автоматичне оцінювання: більшість систем порівнює відповіді користувача з відповідями в базі даних і надає автоматичну оцінку;
- підтримка різних типів завдань: включає завдання на вибір правильної відповіді, відкриті питання та завдання, де потрібно ввести математичну формулу або графік;
- обмеження: незважаючи на наявність великої бази завдань, такі системи можуть обмежувати варіативність завдань, оскільки кожне завдання потребує попереднього налаштування.

Недоліками систем із базою задач є:

- обмежена варіативність: завдання не генеруються динамічно, що може призвести до повторення або передбачуваності;
- менш гнучкі в адаптації: системи зазвичай не здатні адаптувати завдання під поточний рівень знань студента в реальному часі;
- висока залежність від попередньої підготовки бази завдань: для покриття всіх можливих варіантів необхідно підготувати великий обсяг завдань вручну.

Сучасне програмне забезпечення для вивчення математики допомагає не лише виконувати складні обчислення, але й глибше розуміти різні математичні концепції. Серед найбільш популярних інструментів у цій сфері можна виділити MATLAB і Maple, які забезпечують можливості для роботи з символьними та числовими обчисленнями.

MATLAB використовується для математичних обчислень, у тому числі - похідних функцій. Його можливості охоплюють символьне диференціювання використовуючи Symbolic Math Toolbox, чисельне обчислення похідних складних функцій, а також побудову 2D та 3D графіків для наочного представлення результатів. Головними перевагами є широкий набір математичних функцій і модулів, а також можливість інтеграції з іншими програмами, такими як Simulink. Проте вартість ліцензії досить висока, а для ефективного використання необхідні базові знання програмування.

Maple – це система, що підтримує символьні обчислення та використовується для роботи математичними виразами, включаючи диференціювання, інтегрування та аналіз функцій. Інтерактивний інтерфейс спрощує навчання і виконання обчислень, а об'ємна бібліотека математичних функцій і шаблонів допомагає швидко знаходити шукані результати. Однією з ключових переваг Maple є можливість покрокового пояснення результатів, що особливо корисно для студентів і викладачів. Також Maple підтримує як аналітичні, так і чисельні методи обчислення. Серед її недоліків висока вартість ліцензії та обмежена інтеграція з іншими програмами.

Аналіз сучасних досліджень та існуючих інформаційних систем для контролю знань з математичних дисциплін у дистанційній освіті показує, що більшість таких рішень базуються на використанні статичних баз завдань, що кардинально відрізняється від запропонованого підходу.

**Мета дослідження** полягає в розробці та тестуванні алгоритмів генерації задачі, обчислення її розв'язку, порівнянні його з розв'язком, якій одержаний здобувачем, та висновку про вірну (або невірну) відповідь.

**Викладання основного матеріалу дослідження.** Сформулюємо основні вимоги, яким, на думку авторів повинна задовольняти сучасна система контролю знань.

1. Наявність баз завдань та відповідей потребує постійного їх оновлення, що не виключає ймовірності помилок при вводі даних. Збій позиціонування між відповідними елементами масивів завдань та відповідей також буде призводити до невірного оцінювання знань здобувачів. Таким чином, людський фактор буде негативно впливати на надійне функціонування системи.

Основний висновок з цього - система не повинна містити бази завдань та бази відповідей. Завдання повинні генеруватися автоматично на підставі відповідних правил, перевірятися на коректність та вирішуватися в окремому програмному модулі.

На основі цих критеріїв були вирішено розробляти систему у вигляді вебдодатку RESTful API архітектури. Такий підхід забезпечує гнучкість і масштабованість системи, оскільки відсутність статичних баз даних дозволяє динамічно формувати завдання та автоматизувати їх перевірку. Для реалізації цього підходу було обрано архітектуру RESTful API, яка є широко визнаним стандартом у веброзробці. У розроблюваній си-

стемі RESTful API буде відповідати за генерацію завдань, перевірку їхньої коректності та обробку відповідей користувачів. Це забезпечить ефективну взаємодію між клієнтською та серверною частинами, а також дозволить масштабувати систему відповідно до зростаючих вимог.

2. Оскільки мова йде про математичні дисципліни, то відповідь на завдання може бути у вигляді числа, як у більшості існуючих систем, або у вигляді формули. Якщо відповідь розглядається як число, то здобувач повинен обрати одне з декількох варіантів (з яких лише одна вірна відповідь), або ввести власний розв'язок задачі. В останньому випадку висновок про вірну відповідь приймається на підставі оцінки близькості двох чисел – еталонного розв'язку, якій міститься в базі відповідей, та числа, що надає здобувач. Така ідеологія має, окрім чисто технічних проблем, суттєвий недолік, а саме, вона дає можливість здобувачу ввести умовно вірну відповідь не розв'язуючи задачу. Наприклад, в розділі «Диференціальне числення» завдання передбачає обчислити похідну деякої функції  $y = f(x)$  в точці  $x = x_0$ . Якщо здобувач пам'ятає визначення похідної, то він легко знайде її приблизне значення:

$$y' \approx \frac{f(x_0 + \delta) - f(x_0)}{\delta}$$

причому у переважній більшості випадків її точність буде цілком задовільною у порівнянні з результатом аналітичного диференціювання. Очевидно, що для одержання такого розв'язку здобувачу не потрібні знання про правила диференціювання та похідні елементарних функцій. Тим не менш, така відповідь буде інтерпретована як вірна, що не буде об'єктивною оцінкою.

Для запобігання таких випадків система повинна оцінювати відповідь, яка надається не числом, а формулою. Для цього були використанні такі бібліотеки як MathJax [7].

MathJax - це бібліотека для відображення математичних формул на веб-сторінках. Вона дозволяє автоматично перетворювати математичні формули, написані в різних форматах, в графічний вигляд, який можна відображати на сторінці. Вона дозволяє відображати формули в звичному виді, що полегшує сприйняття математичного контенту користувачами.

3. Слід мати на увазі, що в загальному випадку не існує єдиного варіанту записи формули, тобто один і той самий розв'язок задачі можна записати в різний спосіб. Нехай завдання є в визначенні похідної функції

$$y = 3x^2 + 15x + \sin x$$

Вірну відповідь можна представити такими формулами:

$$6x + 15 + \cos x;$$

$$\cos x + 3(2x + 5);$$

$$3(5 + 2x) + \cos x, \text{ і так далі. Всі ці формули є вірними.}$$

Таким чином, інтерпретувати розв'язок у вигляді формули як символну константу абсолютно неможливо. Потрібен алгоритм, який дозволив би зробити висновок про вірну відповідь при наявності значної кількості варіантів запису формули. Тому було розроблено функцію який переводить формулу з текстового типу у математичний тип, що дозволяє проводити математичні операції над виразом.

Резюмуючи вищевикладене, сучасна інтерактивна система контролю знань з математичних дисциплін повинна задовольняти таким вимогам:

- генерація завдань відбувається на підставі правил з використанням випадкових чисел;
- відбувається оцінка відповіді, яка задається формулою;
- висновок про вірну відповідь є об'єктивним та не залежить від способу запису формули.

Нижче будуть розглянути можливі шляхи для забезпечення таких вимог.

**Генерація завдань.** Як приклад розглянемо алгоритм генерації завдань для першого (найнижчого) рівню розділу «Диференціальне числення», який передбачає лише знання здобувача про похідні елементарних функцій та використання основних правила диференціювання.

Чергове завдання генерується таким чином:

$$f(x) = A_1 + A_2, \quad (1)$$

де  $A_1$  та  $A_2 \in [1, 2, 3, 4]$  – натуральні випадкові числа ( $A_1 \neq A_2$ ), які визначають функцію:

- 1 – степенева функція:  $ax^n + b$ ;
- 2 – логарифмічна функція:  $\log_p(cx)$ ;
- 3 – показникова функція:  $q^{dx}$ ;
- 4 – тригонометрична функція:  $\varphi_i(gx)$ .

Параметри функцій обираються випадковим чином як елементи векторів відомої розмірності, наприклад:

$$\bar{a}^T = \bar{d}^T = [-2; -1; 1; 2; 3; 4], \quad \bar{b}^T = [-3; -2; -1; 0; 1; 2; 3; 4], \quad \bar{n} = [1; 2; 3; 4],$$

$$\bar{p}^T = [2; e; 3; 10], \quad \bar{c}^T = \bar{g}^T = [2; 3; 4; 5; 6], \quad \bar{i}^T = [1; 2; 3; 4].$$

Тригонометричні функції визначаються в такий спосіб:

$$\varphi_1 = \sin(gx),$$

$$\varphi_2 = \cos(gx),$$

$$\varphi_3 = tg(gx),$$

$$\varphi_4 = ctg(gx).$$

Таким чином, генерація завдання передбачає наявність шести або семи випадкових чисел для визначення конкретних функцій та їх параметрів. Наприклад, випадковим числом 1; 4; 4; 2; 4; 2; 5 відповідає такий варіант задачі:

$$f(x) = 2x^4 - 2 + \cos(3x) \quad (2)$$

**Алгоритм перевірки відповідей.** Розглянемо ситуацію, коли три здобувача одержали завдання у вигляді (2) та надали такі відповіді:

$$\phi_1(x) = 8x - 3\sin(3x), \quad (3)$$

$$\phi_2(x) = 8x + \sin(3x), \quad (4)$$

$$\phi_3(x) = 8x - \sin(3x) \cdot 3 \quad (5)$$

Очевидно, що функція (4) не є вірним розв'язком задачі, а функції (3) та (5) – вірні розв'язки незважаючи на те, що мають різний вигляд. Очікується, що система спроможна ідентифікувати будь-яку відповідь як вірну, або невірну.

При розробці системи контролю знань використовувався наступний підхід. Функція, зображення якої одержує здобувач, та функція, зображення якої він надає як відповідь, є процедурами-функціями програмного модуля, тобто перетворюють вхідне значення аргументу у вихідне значення функції. Для ідентифікації відповіді як вірної необхідно, щоб приблизне значення похідної функції (2) в будь-якій точці з області існування функції було в деякому сенсі близьким до значення похідної, що визначена з використанням аналітичного диференціювання, тобто зі значеннями функцій (3) – (5) у тій самій точці:

$$\left| \frac{f(x_0 + \delta) - f(x_0)}{\delta} - \phi(x_0) \right| \leq \varepsilon \quad (6)$$

З метою підвищення надійності ідентифікації вірних відповідей вважається доцільним перевіряти умову (6) на множині точок:

$$\sqrt{\sum_{i=1}^k \left( \frac{f(x_i + \delta) - f(x_i)}{\delta} - \phi(x_i) \right)^2} \leq \varepsilon \quad (7)$$

На основі опису можна створити концепт системи для автоматизованого генерування та перевірки знань з теми похідних функцій можна створити UML діаграму (рис. 1):

- генерація випадкових чисел — система на початковому етапі генерує набір випадкових чисел, які будуть використані для створення математичного виразу;
- генерація математичної функції — на основі згенерованих чисел система створює математичний вираз за певними правилами за темою «Диференціальне числення»;
- форматування виразу за допомогою MathJax — математичний вираз передається в модуль MathJax для коректного відображення. Це дозволяє представити його у зручному для читання вигляді на вебсторінці;
- вебсторінка — після того як вираз відформатований і відображений, користувач взаємодіє з вебсторінкою. Він отримує завдання у вигляді математичного виразу та

вводить свою відповідь у поле введення. Для зручності користувач може скористатися набором заготовлених функцій, які дозволяють швидко додати їх до поля введення. Після введення відповіді користувач надсилає свою відповідь до системи для перевірки. Крім того, нижче кнопки відображається відповідь, яка введена користувачем, з метою перевірки коректності свого запису перед надсиланням (рис. 2);

- чисельне диференціювання функції — для кожної згенерованої функції обчислюється чисельна похідна методом скінченних різниць на множині точок;

- обчислення аналітичної похідної — використовуючи аналітичні методи, обчислюється точне значення похідної на множині точок;

- порівняння результатів — порівняння результатів чисельного та аналітичного диференціювання за формулою (7);

- висновок про успішне виконання — на основі порівняння система визначає, чи є відповідь користувача правильною або неправильною, і відображає результат у бінарному виді, або записує його.

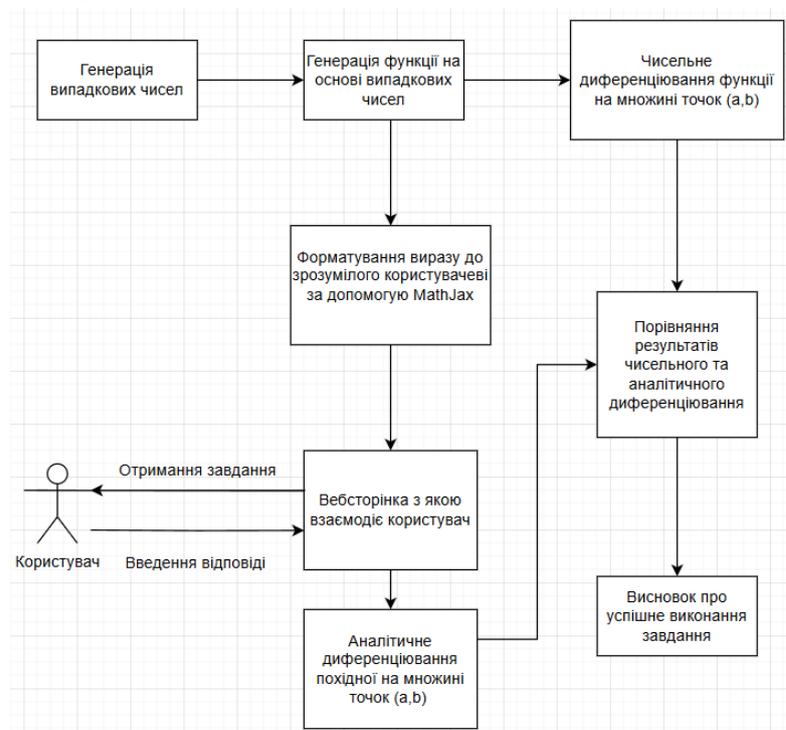


Рисунок 1 – UML діаграма системи



Рисунок 2 – Прототип системи

**Висновки.** Аналіз результатів чисельних експериментів дозволив зробити висновок про доцільність подальшого розвитку систем контролю знань в напрямку, що запропонований авторами. Наявність системи правил дозволяє генерувати значно більшу кількість завдань, ніж при ручному формуванні бази завдань. Можливість надання здобувачем відповіді у вигляді формули та використання авторського алгоритму її перевірки дозволяє більш об'єктивно оцінити рівень знань.

Недоліком розробленої системи є обмеження на складність формул, які має ввести здобувач після виконання завдання. При підвищенні рівня складності підвищується ймовірність помилок при вводі. Таки обмеження виключають можливість генерації задач високого рівня (так званих «задач з зірочкою»).

#### ЛІТЕРАТУРА/ REFERENCES

1. Filatov H.V., Radul O.A. On the implementation of distance learning forms in the educational process of engineering disciplines // Education and Management. Vol. 12. – 2009. – No. 3–4. – P. 118–125.
2. Moodle Course: Junior Mathematics | Mount Orange School.  
URL: <https://school.moodledemo.net/course/view.php?id=46> (date of access: 06.09.2024).
3. Khan Academy. Derivatives course. URL: <https://www.khanacademy.org/math/calculus-1/cs1-derivatives-definition-and-basic-rules> (date of access: 09.09.2024).
4. ALEKS. Adaptive mathematics courses.  
URL: [https://www.aleks.com/about\\_aleks/course\\_products?cmscache=detailed&detailed=ghighedmath16\\_colalgebra#ghighedmath16\\_colalgebra](https://www.aleks.com/about_aleks/course_products?cmscache=detailed&detailed=ghighedmath16_colalgebra#ghighedmath16_colalgebra) (date of access: 12.10.2024).
5. Proctorio. URL: <https://proctorio.com/> (date of access: 12.10.2024).
6. ExamSoft. URL: <https://examsoft.com/> (date of access: 02.11.2024).
7. MathJax Documentation. URL: <https://docs.mathjax.org/en/latest/> (date of access: 02.11.2024).
8. Sobchuk V.V., Chychurin O.V., Kalchuk I.V., Zhygallo T.V. Solving Problems in Analysis and Differential Equations Using Mathematica Computer Algebra Tools: Textbook. Kyiv: Millennium, 2021. 420 p.
9. Andrunyk V.A., Vysotska V.A., Pasichnyk V.V., Chyrun L.B., Chyrun L.V. Numerical Methods in Computer Science: Study Guide. Lviv: Novyi Svit – 2000, 2020. 536 p.

Received 24.01.2026.  
Accepted 29.01.2026.

#### *Concepts for designing knowledge assessment systems in mathematical disciplines under distance education conditions*

*The article presents a novel approach to the design of knowledge assessment systems for students studying the discipline of Differential Calculus. In contrast to most existing systems, which rely on fixed databases of tasks and answers, the proposed solution introduces a dynamic model that automatically generates tasks based on predefined rules using random numbers. Each generated task represents a mathematical function for which the derivative must be calculated. Before presenting the task to the student, the system checks its validity. The answer is submitted not as a numeric value but in the form of a symbolic mathematical expression. This significantly enhances the quality of assessment by enabling evaluation of*

*both correctness and depth of understanding of differentiation methods. Recent advances in symbolic computation, numerical methods, and machine learning have opened new possibilities for developing intelligent systems capable of adapting to the individual needs of learners. Existing platforms — such as Moodle, Khan Academy, ALEKS, Proctorio, and ExamSoft — offer various testing frameworks. However, they are primarily based on static question banks, which often limits their flexibility, adaptability, and ability to provide personalized feedback. This paper outlines the limitations of such systems and highlights the necessity for more adaptive and interactive approaches in mathematical e-learning environments.*

*The objective of the study is to develop and test an algorithm for task generation, automatic solution computation, comparison with the student's submitted answer, and determination of the answer's correctness. The study also aims to ensure that the method used allows for formula-based responses to be evaluated fairly, even when the same solution is presented in multiple algebraically equivalent forms. The core of the proposed system is built on a RESTful API architecture. This allows tasks to be generated in real-time using randomized parameters and validated programmatically before delivery. The solution also utilizes MathJax for formatting and displaying mathematical content in a clear, readable manner within a web interface. An essential component of the system is its capability to interpret symbolic expressions and compare them analytically and numerically. The comparison is done by evaluating the difference between the user-submitted derivative and the system-generated solution over a range of values using both symbolic differentiation and numerical finite difference methods.*

*A specific example is discussed involving the generation of derivative problems from randomly selected elementary functions—polynomial, exponential, logarithmic, and trigonometric. The structure of such problems is defined by a set of randomly generated parameters, which enables a practically unlimited variety of tasks. The system includes mechanisms to compare student-submitted expressions with the reference solution through evaluation at multiple points, reducing the likelihood of false positives and ensuring objectivity. The findings suggest that this rule-based task generation approach enables a higher degree of individualization and significantly reduces the dependency on static databases. Moreover, assessing symbolic expressions rather than mere numerical values allows the system to better reflect the actual understanding and application of calculus concepts by students. The proposed system was prototyped and visualized through a UML diagram and web interface, demonstrating its practicality and user-centered design. The results of numerical experiments support the viability of the proposed approach. The system's rule-based task generation and symbolic response evaluation enhance assessment reliability and objectivity. Unlike systems reliant on preloaded question banks, the proposed method provides flexibility, reduces manual workload, and supports deeper learning. However, the current system is best suited for problems of basic to moderate complexity, as higher-level tasks may require more sophisticated symbolic input and parsing mechanisms.*

*Keywords: distance learning, knowledge control system, differential calculus, system of rules for generating tasks, algorithm for evaluating answers.*

**Зеленцов Дмитро Гегемонович** – д.т.н., професор, завідувач кафедри інформаційних систем, Навчально-науковий інститут “Український державний хіміко-технологічний університет”.

ORCID: <https://orcid.org/0009-0009-1959-3153>

**Шаповалов Кирило Ігорович** – аспірант кафедри інформаційних систем, Навчально-науковий інститут “Український державний хіміко-технологічний університет”.

ORCID: <https://orcid.org/0009-0007-8069-8136>

**Zelentsov Dmytro** – doctor of technical sciences, professor, head of department of information systems, scientific and educational institute “Ukrainian State University of Chemical Technology”.

ORCID: <https://orcid.org/0009-0009-1959-3153>

**Shapovalov Kyrylo** – postgraduate student, department of information systems, scientific and educational institute “Ukrainian State University of Chemical Technology”.

ORCID: <https://orcid.org/0009-0007-8069-8136>