

## ВІРТУАЛЬНЕ НАВЧАЛЬНЕ СЕРЕДОВИЩЕ УЧНІВ ПОЧАТКОВИХ КЛАСІВ З ПІДТРИМКОЮ АВТОМАТИЧНОЇ ГЕНЕРАЦІЇ ТЕСТОВИХ ЗАВДАНЬ

Ленцик І.Ю.<sup>1)</sup>, Августин Р.Р.<sup>2)</sup>, Крепич С.Я.<sup>3)</sup>  
 Тернопільський національний економічний університет  
<sup>1)</sup> магістрант; <sup>2)</sup> к.е.н., доцент, <sup>3)</sup> викладач

### I. Постановка проблеми

Сучасні реформи освіти передбачають підвищення якості, доступності навчання і виховання людини – творчої, ініціативної, самодостатньої. Ми живемо у непростий час, тому наше життя спонукає нас до пошуку ефективніших способів вирішування проблем, розвитку уміння мислити та правильно планувати хід дій. Так як українська освіта орієнтується на європейські стандарти, то великого значення набуває впровадження нових підходів для оцінювання навчальних досягнень.

Однією із найважливіших складових навчання є контроль знань учнів. Під словом "контроль" розуміється відношення одержаних результатів із цілями, які були заплановані.

Саме тому, залишається актуальним питання щодо створення такого віртуального навчального середовища, яке дозволить вчителю провести комп'ютерне тестування для об'єктивного оцінювання вмінь та знань учнів. Комп'ютерне тестування дозволяє втілити основні принципи контролю навчання: системності перевірки, оцінки вмінь і знань; принцип об'єктивності та принцип однаковості вимог.

### II. Мета роботи

Метою роботи є розробка віртуального навчального середовища з підтримкою автоматичної генерації тестових завдань для задач з математики початкової школи з керованою складністю, яка дозволить вчителю дистанційно, швидко та об'єктивно провести перевірку вивченого матеріалу учнями.

### III. Метод генерування тестів

В основу роботи віртуального навчального середовища покладений принцип керованої складності [1]. Метод, який буде використовуватись при автоматичній генерації тестових завдань для учнів – це метод із програмованим оператором, де в якості оператора буде виступати алгебраїчний вираз. Складність тестового завдання залежить від діапазону значень, які прийматиме певний вираз. Якщо діапазон значень менший, то складність менша, а якщо діапазон значень ширший, то відповідно складність більша. Тому кількість вхідних параметрів у випадку з програмованим оператором буде обмежена. Метод генерації задач із програмованим оператором передбачає генерацію завдання на природній мові для учня та відповідне завдання для інтерпретатора.

Отже, умова задачі буде генеруватися так:

$$\text{Task} = \langle \text{TaskS}, \text{TaskI} \rangle, \quad (1)$$

де TaskS – це умова завдання для учня, а TaskI – це умова завдання для інтерпретатора відповідно.

Формуємо далі умову задачі для учня на природній для нього мові:

$$\begin{aligned} \text{TaskS\_C} &= \pi_{\text{IdFt}, \text{IdCh}, \text{Lg\_Int}, \text{Tcv}, \text{IdTf}, \text{IdVtv}} \\ &(\sigma_{\text{IdCh}=\text{IdCh\_C}}(\tau_{\text{RANDLIMIT}(1)}(\text{Ft}))). \end{aligned} \quad (2)$$

Доповнюємо умову задачі вхідними змінними та уточнюємо умову задачі, після чого виводимо умову завдання для студента на екран.

Коли завдання для учня сформульовано, генеруємо завдання для інтерпретатора:

$$\begin{aligned} \text{TaskI\_C} &= \pi_{\text{IdFt}, \text{IdCh}, \text{Nmf}, \text{Tcv}, \text{IdTf}, \text{IdVtv}} \\ &(\sigma_{\text{Ft}(\text{IdFt})=\text{TaskS\_C}(\text{IdFt})}(\text{Ft}, \text{TaskS\_C})). \end{aligned} \quad (3)$$

Доповнюємо умову задачі контрольними вхідними змінними, уточнюємо умову завдання для інтерпретатора та фіксуємо результат виконання завдання інтерпретатором. Якщо результат розв'язку умови задачі учнем співпадає із результатом розв'язку завдання інтерпретатором, то завдання розв'язане правильно [1].

Тестове завдання складатиметься із запитання та чотирьох відповідей, одна з яких є правильною. В кінці кожного тесту у дужках буде вказаний тип складності завдання, тобто на який рівень воно розраховано чи це низький, середній, достатній чи високий. Для проходження тестування учневі необхідно зареєструватись та авторизуватись у системі. Після чого він матиме змогу пройти тестування, а також запропонувати свій приклад тесту, який зможуть спробувати пройти його однокласники. Після тестування результат проходження з'явиться на екрані.

#### IV. Проектування та реалізація віртуального навчального середовища

У результаті проведених досліджень, розроблено віртуальне навчальне середовище на основі веб-технологій. В ролі програмованого оператора використовується інтерпретатор мови Spring Expression Language, що є частиною Spring Framework для Java Enterprise Edition. Тести будуть описані у формі XML, так як формат XML має безліч переваг серед яких:

- стиснутість файлів. Файли автоматично стискаються та іноді можуть ставати на 75 процентів менші,
- покращене відновлення пошкодженого файлу. Файли мають модульну структуру, яка дозволяє різним компонентам даних у файлі зберігатись окремо один від одного,
- підвищена конфіденційність і більше можливостей керування особистими відомостями,
- можливість багаторівневого використання елементів програмованих операторів, в яких один елемент може включати в себе довільну кількість вкладених.

Збереження даних у XML-форматі дозволить вчителям обмінюватись файлами із тестовими завданнями, модифікувати їх та вдосконалювати. Вчителі матимуть змогу обробляти записи у XML-форматі за межами системи. Непотрібно додатково встановлювати спеціальне програмне забезпечення. Основною перевагою збереження файлів у форматі XML є зручність супроводу їх у подальшому.

Для проходження тестових завдань учню необхідно зареєструватись у системі, пройти авторизацію та обрати категорію тестування. Результати проходження тестування будуть зберігатись у системі та будуть доступні лише вчителю. Приклад тестового завдання наведено на рисунку 1.

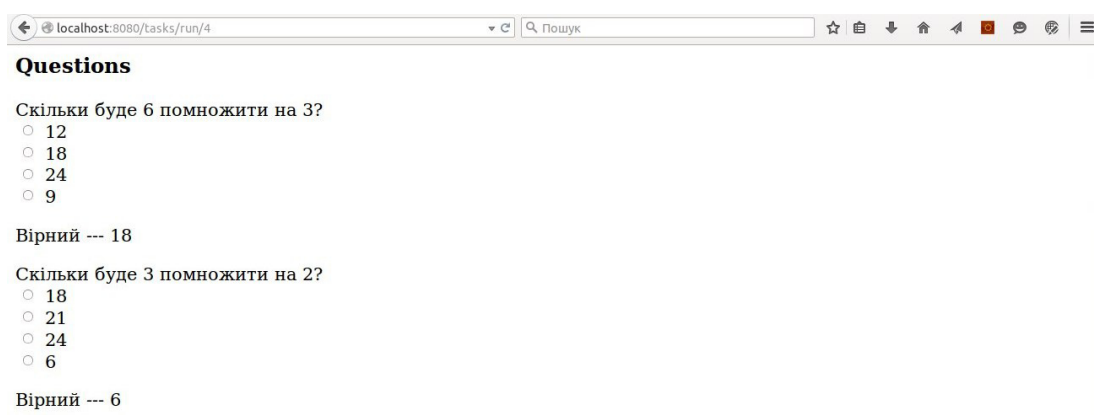


Рисунок 1 - Приклад тестового завдання для учнів

#### Висновок

Розроблено віртуальне навчальне середовище, яке надає можливість вчителю формувати тестові завдання різного рівня складності. Вчитель зможе легко здійснювати модифікацію тестового завдання, так як для цього не потрібно встановлювати спеціальне програмне забезпечення. Система дозволить учням проходити тести, результати яких зберігатимуться у системі і будуть доступні для перегляду лише вчителю. Розроблена система забезпечує швидкий, надійний та якісний спосіб контролю знань учнів.

#### Список використаних джерел

1. Мельник, А. Інформаційна технологія автоматичної генерації тестових завдань з керованою складністю [Електронний ресурс] / А.М. Мельник, Р.М. Пасічник, Р.П. Шевчук // Системи обробки інформації. – 2011. - Вип. 3 (93). – С. 57-61.
2. Мельник А.М. Автоматична генерація тестових завдань різних типів / А.М. Мельник // Вісник Хмельницького національного університету. – 2010. – № 4. – С. 124- 129. 3
3. Сергушичева А. П. Метод и алгоритмы автоматизированного построения компьютерных тестов контроля знаний по техническим дисциплинам: автореф. дис. на получение ученой степени канд. техн. наук: 05.13.01 / Вологодский государственной технической университет /А. П. Сергушичева. – СПб, 2007. – 18 с.