

12. Энергоснабжение сельскохозяйственных потребителей с использованием возобновляемых источников: учеб. пособие для вузов / Л.А. Саплин, С.К. Шерьязов, О.С. Пташкина-Гирина [и др.]; под общ. ред. Л.А. Саплина. - Челябинск, 2000. - 194 с.

13. Охоткин Г.П., Серебрянников А.В. Основные принципы построения автономных солнечных электростанций [Электронный ресурс] // Современные проблемы науки и образования. 2012. №6. URL: <http://www.science-education.ru/106-7345> (дата обращения: 09.11.2012).

14. Шерьязов С.К. Обоснование эффективной системы энергоснабжения с использованием возобновляемой энергии / С.К. Шерьязов // Ползуновский вестн. АлтГТУ. - 2006. - Вып. 4. - №2. - С. 434-439.

15. Щекун Л. Сонячна енергетика освоєє південні степи / Л. Щекун // Уряд. кур'єр. – 2012. – 23 лист.(№ 216). – С 11.

Рукопис подано до редакції 02.04.2019

УДК 681.03

І.Н. ВДОВИЧЕНКО, канд. техн. наук, доц., О.С. ВАТРЯ, ст. викладач  
Криворізький національний університет

## АНАЛІЗ ДОДАТКУ ЕМУЛЯЦІЇ ТРИВИМІРНОЇ СЦЕНИ ІЗ ВИКОРИСТАННЯМ ШЕЙДЕРІВ ТА ЗАСОБІВ OPENGL

У публікації розглянуто спосіб поєднання функціоналу OpenGL з засобами шейдерів для досягнення оптимальної швидкодії при обробці графічних об'єктів.

**Мета дослідження:** побудова тривимірної сцени за допомогою шейдерів та бібліотеки OpenGL.

**Методи дослідження:** програмування та математична обробка наборів векторних графічних примітивів.

**Наукова новизна:** системний аналіз побудови тривимірної сцени за допомогою шейдерів та бібліотеки OpenGL.

**Практична значимість:** удосконалено можливості відкритого стандарту, такі як поглинання та розсіювання світла, накладення текстур та ефекту пост-обробки. Представлені блок-схема алгоритму роботи шейдерів вершин та геометрії та **блок-схема алгоритму роботи шейдерів вершин та геометрії для створення пост-обробки**. Відзначено особливості програмування даної сфери. Запропоновано методи та алгоритми вирішення задач створення програмного забезпечення візуалізації тривимірної сцени.

**Результати:** у статті представлено можливості додатку для моделювання тривимірної сцени із використанням потужностей шейдерів та OpenGL. Розглянуто призначення шейдерів. Відмічено, що OpenGL - це графічний стандарт в області комп'ютерної графіки. На даний момент він є одним з найпопулярніших графічних стандартів у всьому світі. Стаття характеризує OpenGL, як єдиний стандарт для розробки тривимірних додатків, що поєднує в собі такі якості як силу і в той же час простоту. Мультиплатформеність дозволяє легко переносити програмне забезпечення з однієї операційної системи в іншу. Активне використання тривимірної графіки в багатьох сферах наукової діяльності та промисловості є актуальною проблемою. Традиційно підтримка тривимірної графіки обмежувалася високопродуктивними комп'ютерами або спеціалізованими ігровими консолями, а її програмування вимагало застосування складних алгоритмів. Однак, завдяки зростанню продуктивності персональних комп'ютерів і розширенню можливостей браузерів стало можливим створення і відображення тривимірної графіки із застосуванням нових технологій.

**Ключові слова:** шейдер, тривимірні сцени, графічні об'єкти, інтерфейс, відеоефекти.

doi: 10.31721/2306-5451-2019-1-48-154-159

**Об'єктом дослідження** є шейдери GLSL та OpenGL, їх зв'язок.

**Предметом дослідження** є застосування шейдерів та бібліотеки OpenGL для відображення графічних об'єктів.

**Аналіз досліджень і публікацій.** Розробкою та використанням OpenGL займаються найбільші фірми розробники як обладнання так і програмного забезпечення: Silicon Graphics, Inc., Microsoft, IBM Corporation, Sun Microsystems, Inc., Digital Equipment Corporation (DEC), Evans & Sutherland, Hewlett-Packard Corporation, Intel Corporation та Intergraph Corporation. Програмісти, дизайнери, розробники сайтів використовують можливості бібліотек OpenGL, тому тема, що розглянута автором досить актуальна[5].

**Постановка задачі.** Побудова тривимірної сцени за допомогою шейдерів та бібліотеки OpenGL, з розробленням наступних можливостей: поглинання та розсіювання світла; накладення текстур; ефект пост-обробки; завантаження тривимірної моделі за допомогою бібліотеки Assimp.

**Викладення матеріалу та результати. Основним принципом** роботи OpenGL є отримання наборів векторних графічних примітивів у вигляді точок, ліній та багатокутників з наступною математичною обробкою отриманих даних та побудовою растрової картини на екрані і/або в пам'яті. Векторні трансформації та розстеризація виконуються графічним конвеєром (graphics pipeline), який власне являє собою дискретний автомат. Абсолютна більшість команд OpenGL потрапляють в одну з двох груп: або вони додають графічні примітиви на вхід в конвеєр, або конфігурують конвеєр на різне виконання трансформацій.

OpenGL є низькорівневим процедурним API, що змушує програміста диктувати точну послідовність кроків, щоб побудувати результуючу растрову графіку (імперативний підхід). Це є основною відмінністю від дескрипторних підходів, коли вся сцена передається у вигляді структури даних (найчастіше дерева), яке обробляється і будується на екрані.

З одного боку, імперативний підхід вимагає від програміста глибокого знання законів тривимірної графіки та математичних моделей, з іншого боку — дає свободу впровадження різних інновацій.

Програмовані шейдери гнучкі та ефективні. Складні на вигляд поверхні можуть бути візуалізовані за допомогою простих геометричних форм. Наприклад, шейдери можуть бути використані для малювання поверхні із тривимірної керамічної плитки на абсолютно плоскій поверхні.

В програмних графічних рушіях весь ланцюжок рендерингу — від визначення видимих частин сцени до накладення текстури — створювався розробником ігор. У цей ланцюжок можна було включати власні нестандартні відеоефекти. З появою перших відеоакселераторів розробник виявився обмежений тим набором ефектів, який закладений в апаратне забезпечення.

Шейдери були розроблені для того, щоб мати змогу відтворювати складні відеоефекти з атомарних операцій. Попередниками шейдерів були процедурна генерація текстур (що широко застосовувалася в Unreal для створення анімованих текстур води і вогню) і мультитекстурування. Але і ці механізми не забезпечують такої гнучкості, як шейдери.

Задачі створення програмного забезпечення візуалізації тривимірної сцени були поставлені наступні вимоги:

- зрозумілий інтерфейс та керування між користувачем та програмним забезпеченням;
- використання актуальних можливостей системи;
- відображення результату роботи програми у вікні, що створене за допомогою WinAPI;
- завантаження моделі будь-якого формату у програму та її візуалізація;
- підтримка зміни ефектів пост-обробки у режимі роботи програми;
- достовірна робота світла із геометрією сцени.

Вихідна інформація представляє собою результат роботи об'єкту OpenGL framebuffer, який перед передачею у контекст вікна WinAPI було згладжено за допомогою технології MSAA для більш привабливого зображення.

В якості вхідної інформації у програму надходять різноманітні за форматом файли.

Для моделей у даному додатку використовується формат .obj, формат файлу опису геометрії, вперше створений в Wavefront Technologies для їх анімаційного пакету Advanced Visualizer. Це відкритий файловий формат, і він був прийнятий іншими розробниками 3d редакторів, як стандартний. Формат OBJ дуже простий, і задає тільки геометрію об'єкта, а якщо конкретно, то координати кожної вершини, її текстурні координати, нормалі, і грані, що задаються списками вершин багатокутників. Вершини многокутників за замовчуванням задаються проти годинникової стрілки, роблячи явне задання нормалей необов'язковим. Один файл OBJ може посилатись на кілька зовнішніх матеріалів.

Для зображень використовуються всі формати, що підтримує бібліотека SOIL(BMP, JPEG, PNG...).

Далі буде наведено загальну структуру об'єктів додатку, їх використання та властивості.

У програмі реалізовано аналог C# «LayoutManager», функціями якого є динамічна зміна макету вікна, зміна розмірів кожного із дочірніх вікон в залежності від розмірів інших, а також параметрів структур цього класу.

Створений клас «Element» містить структури, такі як: якорь, стикування, орієнтація, також вектор таких же класів «Element» а також вказівку на батька, для того щоб створити зв'язок по типу колекції List.

Таким чином при зміні розмірів одного із вікон посилається повідомлення головному вікну, котре є батьком, в якому робляться перерахунки в залежності від заповнених структур, наприклад відносна чи абсолютна ширина вікна, чи горизонтальна, чи вертикальна орієнтація.

У програмі реалізована клас-обгортка «OpenGLWrapper», котрий потрібен для полегшення роботи програміста із функціями та об'єктами OpenGL. За допомогою цього класу вдалося відділити реалізацію вікон на WinAPI та роботу із шейдерами і OpenGL.

У програмі було реалізовано два класи «Mesh» та «Model», завдання першого є налаштування текстур полігональної сітки «або меша» та генерація масивів вершин та буферів для відображення. Завданням другого є автоматизація роботи із бібліотекою Assimp, тобто завантаження моделі, отримання матеріалів, завантаження текстур із файлів, заповнення структур вершин, нормалей, а також функція відтворення моделі у вікні.

Також реалізовано клас «Shader», призначенням якого є завантаження текстового документу, отримання та компіляція даних та лінування до програмного об'єкту для подальшого використання.

Реалізовано клас «Camera», що відокремив значну частину роботи із переміщенням і трансформацією камери.

У класі «Object» реалізовано клас «GamePool», «SkyBox», «FrameBuffer», «Object3D».

Перший клас реалізований для зручного розміщення тривимірних об'єктів, їх видалення та доступ до них із головного класу програми.

Другий клас відповідає за створення кубу, що імплементує створення Cubemap-у, тобто кубу, що склеює шість послідовних текстур, імітуючи навколишнє середовище.

Третій клас відповідає за ефекти пост-обробки, в даному програмному забезпеченні створено роботу із MSAA сглажуванням об'єктів для відображення більш гладких поверхонь, ефект «чорно-білого екрану», інверсія кольорів.

Четвертий клас, в ньому реалізовано створення примітивів по типу куба чи сфери.

Шейдери надали потужний засіб для реалізації складних тривимірних ефектів, postprocessing-ефектів і збільшення загальної швидкодії. Хоча якісне використання шейдерів - складне завдання, що вимагає наявності безлічі додаткових знань, переваги використання шейдерів помітно перевищують недоліки.

Для аналізу механізму роботи шейдерів наведено декілька алгоритмів, за якими вони функціонують. Наприклад, рис. 1.

*Робота шейдеру вершин та фрагментного шейдеру для роботи зі пост-обробкою*

Для цієї задачі необхідно передати булеві змінні до шейдеру та змінити стандартний фреймбуфер на свій. Для можливості застосування технології MSAA було створено два FrameBuffer-и, один для згладжування, а другий як контейнер кольорів для передачі у шейдер, адже шейдер не може прийняти фреймбуфер згладжуваного типу.

Для передачі інформації с MSAA буферу необхідно зробити такі дії:

```
glBindFramebuffer(GL_READ_FRAMEBUFFER, frame->framebuffer); //Прочитати Буфер;
glBindFramebuffer(GL_DRAW_FRAMEBUFFER, frame->nonmsFramebuffer); //Відкрити
буфер для запису;
glBlitFramebuffer(0, 0, screen.cx, screen.cy, 0, 0, screen.cx, screen.cy,
GL_COLOR_BUFFER_BIT, GL_NEAREST); //Перекинути інформацію.
```

Крім того:

отримати текст шейдерів;

скопіювати шейдери;

розпочати використання програми, до якої під'єднано шейдери;

заповнити uniform-змінні та атрибути шейдеру вершин необхідними даними; Для цього:

передати матриці проєкцій, виду та моделі;

передати позицію вершини до шейдера вершин;

передати нормаль до шейдера вершин;

передати координати текстури до шейдера вершин.

заповнити uniform-змінні та атрибути шейдеру геометрії необхідними даними;

передати позицію камери;

передати Sampler2D текстури;

передати Cubemap;

передати структури направленої світла, типу ліхтарика та місцевого світла.

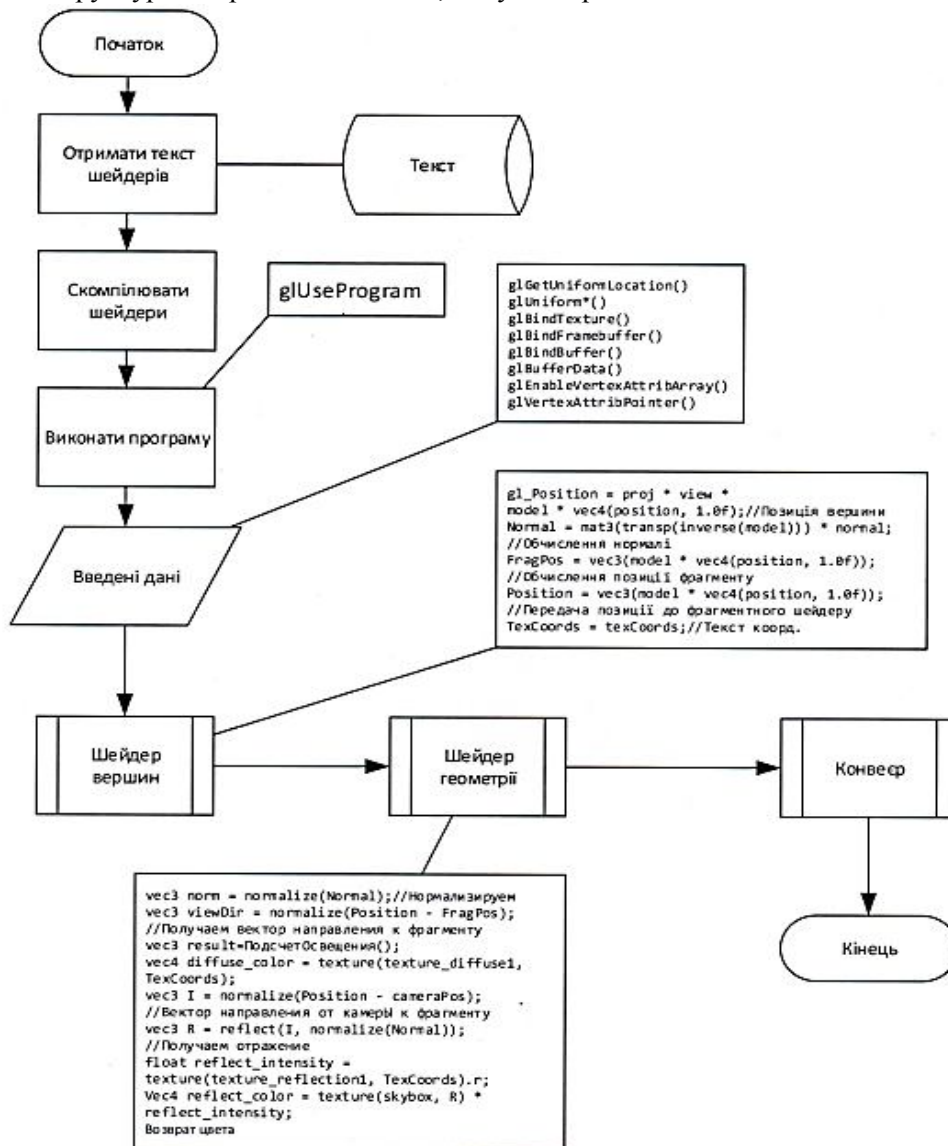


Рис.1. Блок-схема алгоритму роботи шейдерів вершин та геометрії

Програма обчислює прийняті дані та обмінюється між шейдерами згідно до OpenGL конве-  
єру;

виведення результату на екран.

Схема алгоритму роботи програми приведена на рис. 2.

OpenGL має продуману архітектуру і досить простий процедурний інтерфейс. Незважаючи на це, за допомогою цієї графічної бібліотеки можна створювати складні і потужні програмні комплекси, витрачаючи при цьому мінімальний час в порівнянні з іншими графічними бібліотеками.

**Висновки та напрямок подальших досліджень.** Традиційно підтримка тривимірної графіки обмежувалася високопродуктивними комп'ютерами або спеціалізованими ігровими консолями, а її програмування вимагало застосування складних алгоритмів. Однак, завдяки зростанню продуктивності персональних комп'ютерів і розширенню можливостей браузерів стало можливим створення і відображення тривимірної графіки із застосуванням нових технологій [6]. В статті представлено використання сучасних технологій підтримки тривимірної графіки. Як результат, було задіяні наступні технології для реалізації додатку:

здіяні нові ступені обробки шейдерів, що дозволяє перенести обробку теселяції з центрального процесора на GPU;

підтримка вибіркового фрагментних шейдерів (per-sample fragment shaders) і програмованих фрагментних шейдерів вхідних позицій для збільшення якості рендерінга і гнучкості в управлінні згладжуванням;

промальовування даних, що згенеровані OpenGL або такими зовнішніми API, як OpenCL, без участі центрального процесора;

шейдерні підпрограми допоможуть значно збільшити гнучкість написання програм;

розділення стану текстур і текстурованих даних через додавання нового типу даних під назвою «об'єкти семплів»;

64-бітова подвійна точність з плаваючою комою операцій з шейдерами і введенням-виведенням для збільшення точності і якості рендерінга;

збільшення продуктивності, включаючи шейдери геометрії зразка, масивів зразка і нових запитів до таймера.

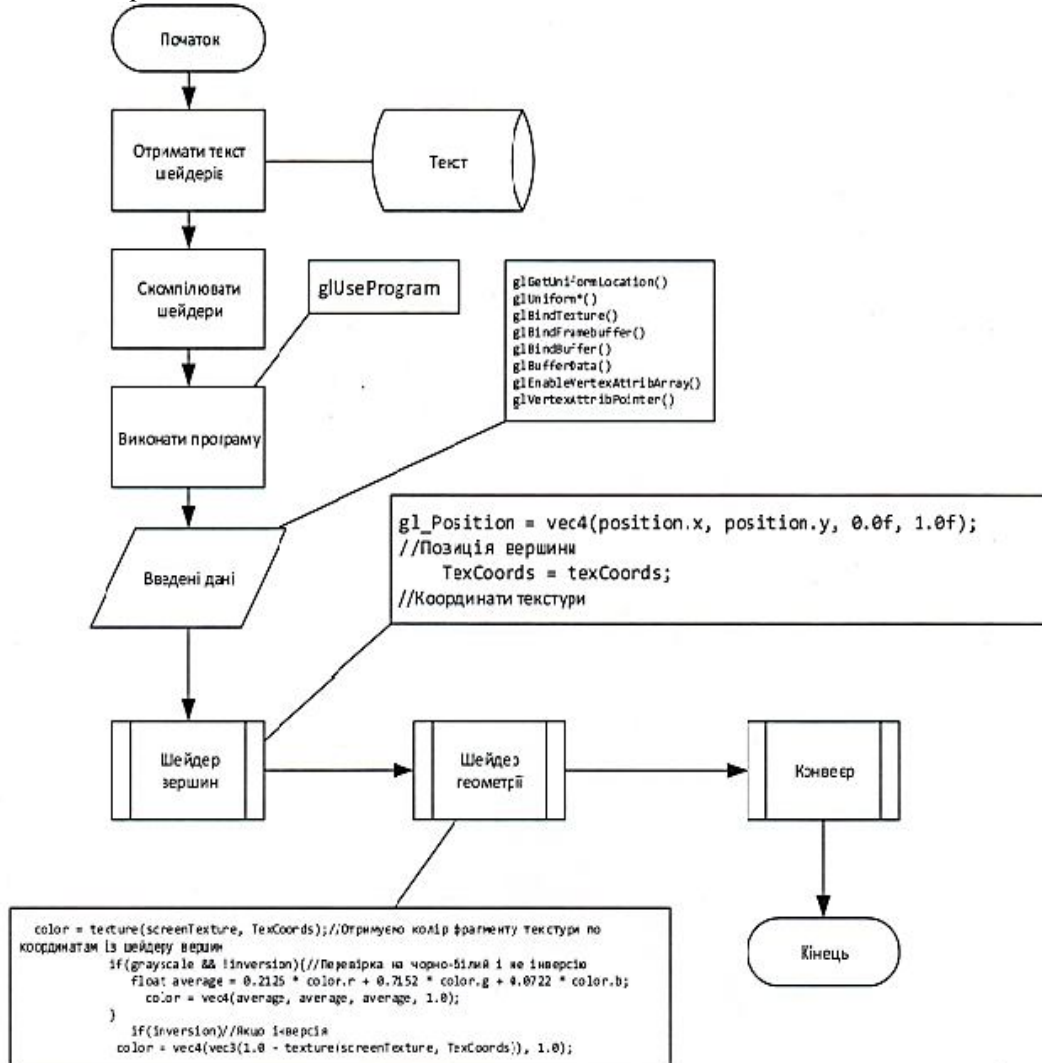


Рис. 2. Блок-схема алгоритму роботи шейдерів вершин та геометрії для створення пост-обробки

### Список літератури

1. Боресков А. В. Расширения OpenGL. — БХВ-Петербург, 2005. — 688с.
2. Боресков Алексей. Разработка и отладка шейдеров. — БХВ-Петербург, 2006. — 488с.
3. Туманов Владислав Застосування сучасних засобів OpenGL в QT. 2016 [Електронний ресурс] – Режим доступу: <https://iconfs.net/infocom2016/>
4. OpenGLintroduction [Електронний ресурс] – Режим доступу: <https://open.gl>
5. <https://3dnews.ru/169184>
6. Коичи Мацуда, Роджер Ли. WebGL: программирование трехмерной графики. /Пер. с англ. Киселев А.Н.-М.: ДМК Пресс, 2015.- 494с.
7. Себастьян Сент-Лаурент. Шейдеры для программистов игр и художников [Електронний ресурс] - Режим доступу: <http://www.netlib.narod.ru/library/book0053/index.htm>

8. Вольф Д. OPENGL 4. Язык шейдеров. Книга рецептов. М.: ДМК Пресс, 2015. – 368с.
9. Горнаков Станислав. Инструментальные средства программирования и отладки шейдеров в DirectX и OpenGL. — БХВ-Петербург, 2005. – 256с.
10. Дэн Гинсбург. OpenGL ES 3.0. Руководство разработчика. М.: ДМК Пресс, 2014. - 449с.

Рукопис подано до редакції 04.04.2019

УДК 581.524.3

К.В. ГЕРАСИМОВА, канд. техн. наук, доц., Криворізький національний університет  
В.В. ТИХОСТУП, ст. викладач, Криворізький коледж Національного авіаційного університету

## **ФІЗИКО-ХІМІЧНІ ПОКАЗНИКИ ВОДИ ШЛАМОСХОВИЩ КРИВОРІЗЬКОГО РЕГІОНУ ТА ЇХ ЗВ'ЯЗОК З ФЛОРИСТИЧНОЮ СТРУКТУРОЮ ПРИБЕРЕЖНОЇ РОСЛИННОСТІ**

**Мета.** Вивчення флористичної структури рослинного покриву шламосховищ Криворізького регіону; визначення екологічної якості води, як формують компоненту прибережних рослинних угруповань.

**Методи дослідження.** Для досягнення поставленої мети використовувались фізико-хімічні методи визначення складу води на вміст металів та йонів. Екологічна оцінка якості води в шламосховищах визначалася на основі загальновідомої класифікації А. А. Алекіна. При вивченні складу рослинних угруповань використовувались загальноприйнятні методики геоботанічних описів. При визначенні видового складу рослинного покриву використовувався «Визначник вищих рослин України».

**Наукова новизна.** Встановлено, що за тривалий час склалися певні біогеоценологічні відносини між елементами техногенно-змінених ландшафтних систем. Серед них важливу роль відіграють прибережні рослинні угруповання, які ще дотепер недостатньо вивчені дослідниками. Визначено, що склад води є формують компонентом прибережного рослинного покриву досліджуваних об'єктів. Проведено фізико-хімічний аналіз води шламосховищ та виявлено прямий зв'язок показників води з флористичною структурою прибережної рослинності.

**Практична значимість.** Отримані результати досліджень дозволяють зробити розрахунок необхідного ступеня очищення стічних вод шламосховищ і мають важливе практичне значення для інженерів-екологів. Ці показники дозволяють правильно обирати методи очищення стічних вод з урахуванням рослинних угруповань, ступеня очищення і рН-стану водного джерела. В техногенних умовах особливо посилюється необхідність аналізу гідробіологічних процесів, що дає можливість прогнозувати зміни якості води і подальший стан існування гідробіоти для попередження небезпечних екологічних наслідків. Ефективне проведення такого аналізу можливе за умови комплексного підходу, а саме – застосування гідрохімічних і біоіндикаційних методів. Вони дозволяють діагностувати зміни екологічного стану природного середовища.

**Результати.** Запропоновано таксономічну, екологічну, біоморфичну та географічну структуру рослинних угруповань шламосховищ Криворізького регіону; сформовані типи рослинних угруповань за гідрохімічними показниками складу води.

**Ключові слова:** шламосховище, аналіз води, прибережна рослинність, екологічна оцінка.

doi: 10.31721/2306-5451-2019-1-48-159-164

**Проблема та її зв'язок з науковими та практичними завданнями.** Упродовж багатьох років багатства надр Криворізького регіону інтенсивно споживались, що призвело до появи ландшафтно-змінених техногенних систем. Процес деградації природних екосистем є наслідком порушення екологічної рівноваги і призводить до руйнації навколишнього середовища. Гірничо-збагачувальні комбінати та шахти є джерелами забруднення довкілля. До їх складу входять шламосховища, які тільки на Криворіжжі займають площу понад 7600 га і є потужними джерелами пилоутворення, спричиняють засолення значних територій, змінюють їх гідрологічний режим [3, 4, 9]. Основними елементами шламосховищ є огорожувальні споруди (дамби), відкоси наміву, ставки освітлення води, на яких під впливом техногенних систем формується гідрофітна, гігрофітна та мезофітна рослинність [2, 3, 5]. За тривалий час склалися певні біогеоценологічні відносини між елементами техногенно-змінених ландшафтних систем, серед яких важливу роль відіграють прибережні рослинні угруповання, які досі ще недостатньо вивчені. Метою даної роботи є вивчення флористичної структури прибережної рослинності, а також екологічна оцінка води шламосховищ, як формують компоненту рослинних угруповань.