

Міністерство освіти і науки України
Чернівецький національний університет
імені Юрія Федьковича

Матеріали

студентської наукової конференції
Чернівецького національного університету
імені Юрія Федьковича

**ФАКУЛЬТЕТ
МАТЕМАТИКИ ТА ІНФОРМАТИКИ**

16-17 квітня 2019 року



Чернівці
Чернівецький національний університет
імені Юрія Федьковича
2019

Матеріали студентської наукової конференції Чернівецького національного університету (16-17 квітня 2019 року). Факультет математики та інформатики. – Чернівці : Чернівец. нац. ун-т ім. Ю. Федьковича, 2019. – 66 с.

До збірника увійшли статті студентів факультету математики та інформатики, підготовлені до щорічної студентської наукової конференції університету.

Молоді автори роблять спробу знайти підхід до висвітлення й обґрунтування певних наукових питань, подати своє бачення проблем.

© Чернівецький національний
університет імені Юрія Федьковича, 2019

Створення блокчейн додатку на базі платформи Ethereum

Технологія блокчейн поступово змінює світ так само сильно, як це зробив Інтернет. Інтерес до цієї технології зростає з кожним днем і незабаром багато індустрій будуть користуватися нею. Блокчейн (Blockchain)— це розподілена база даних, що складається з ланцюжків транзакцій. Всі несанкціоновані втручання для зміни інформації в блокчейн-технологіях є неможливими. Технологія була спочатку розроблена для цифрової валюти - біткоїнів, але на даний час технічна спільнота розробляє і втілює інші варіанти.

У процесі розробки створено блокчейн додаток на базі мережі Ethereum [1]. Ethereum - платформа, заснована на блокчейні з можливостями використання смарт контрактів. Внутрішня валюта платформи - це ether або ефір. Скорочене позначення - ЕТН. Творцем і головним ідейним натхненником Ethereum є канадський розробник російського походження Віталій Бутерін. За допомогою цього додатка можна ознайомитись з новими технологіями цієї мережі, зрозуміти їхню роль в сучасному програмуванні та в майбутньому.

Спершу було створено смарт контракт. Його сутність полягає в тому, що серед всіх працівників певного закладу обирається директор методом голосування. Кожний працівник може подати свою кандидатуру та голосувати за себе або за іншого працівника. Цей смарт контракт був написаний за допомогою мови Solidity [2]. Solidity – JavaScript-подібна мова програмування для розробки розумних контрактів для EVM. Solidity дозволяє розробникам розробляти програми, які містять логіку для створення алгоритмів проведення транзакцій в мережі Ethereum.

Наступний крок у розробці додатку є взаємодія з користувачем. Це зроблено за допомогою бібліотеки React [3], так як вона зручна в написанні користувацьких інтерфейсів і легка в

розумінні. React отримує інформацію з HTML сторінки від клієнта, обробляє її і повертає результат.

Для стилізації веб-додатку використано фреймворк Bootstrap [4]. Bootstrap - це набір HTML + CSS інструментів і шаблонів для верстки і більш ефективного і швидкого створення сайтів і веб-додатків.

Web3.js[5] - це бібліотека, яка дозволяє взаємодіяти з мережею Ethereum, використовуючи з'єднання HTTP або IPC. Вона присутня у цьому проєкті для отримання інформації з бази даних Ethereum та для того, щоб надсилати інформацію на ноди.

Додаток працює за допомогою розширення для браузерів Metamask [6]. Metamask - це криптовалютний гаманець, який вбудовується в браузер. Він потрібен для спрощення передачі Ефіру (Ефіріум, Ethereum, ETH) або токенів ERC-20 в мережі Ethereum. За допомогою цього розширення ми можемо бачити та підтверджувати всі транзакції.

Проект створений для дослідження блокчейн технологій, а саме платформи Ethereum, за допомоги якої можна створити практично будь-які децентралізовані онлайн-сервіси. Велику роль приділялось написанню смарт контракту на мові Solidity, взаємодії з користувачем за допомогою бібліотеки React та з базою даних Ethereum за допомогою бібліотеки Web3.

Список літератури

1. Ethereum documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <http://www.ethdocs.org/en/latest/>
2. Solidity documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://solidity.readthedocs.io/en/v0.5.3/>
3. React documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://reactjs.org/docs/getting-started.html>
4. Bootstrap. Bootstrap 4 documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://getbootstrap.com/docs/4.1/getting-started/introduction/>
5. Web3.js. Web3.js 1.0 documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://web3js.readthedocs.io/en/1.0/>
6. Metamask documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://metamask.github.io/metamask-docs/>

Розробка розподіленої системи зберігання та перевірки лабораторних робіт студентів

Ручна перевірка лабораторних робіт студентів вимагає великої кількості часу на введення вхідних даних, порівняння результатів правильними та оцінки усіх тестових випадків. Для уникнення витрат часу можна використовувати автоматичні програми, але існуючі рішення мають велику кількість недоліків. Рішенням цієї проблеми є розподілені системи перевірки з легким процесом налаштування та розгортання.

Розподілені обчислення – спосіб розв'язання трудомістких обчислювальних завдань з використанням двох і більше комп'ютерів, об'єднаних в мережу. Розподілені обчислення є окремим випадком паралельних обчислень, тобто одночасного розв'язання різних частин одного обчислювального завдання декількома процесорами одного або кількох комп'ютерів.

На сьогодні, можна створювати системи для розподілених обчислень на основі єдиного комп'ютеру за допомогою створення віртуальних контейнерів за допомогою системи Docker або аналогічних рішень.

Docker – інструментарій для управління ізольованими контейнерами. Docker дозволяє не переймаючись вмістом контейнера запускати довільні процеси в режимі ізоляції і потім переносити і клонувати сформовані для даних процесів контейнери на інші сервери, беручи на себе всю роботу зі створення, обслуговування і підтримки контейнерів.

Існуючі рішення для автоматичної оцінки надають неточні результати використання часу процесору та оперативної пам'яті, оскільки вони працюють в просторі користувача.

Рішенням проблеми точної оцінки використання ресурсів комп'ютера є використання доступних інтерфейсів самого ядра ОС для вимірювання. ОС Linux надає інтерфейс “cgroup v2”, який

створює віртуальні групи користувачів з будь-якими обмеженнями ресурсів і надає точні дані про їхнє використання.

Основними цілям при розробці є:

- 1) незалежність елементів систем;
- 2) винесення усієї конфігурації в окремі файли;
- 3) автоматичне розгортання системи;
- 4) вертикальне масштабування.

Розроблено 3 додатки системи 'Ligtest', які дозволяють розгорнути усю необхідну інфраструктуру для зберігання, редагування даних про задачі, компіляцію та тестування відправлень користувачів, реєстрацію та авторизацію. Усі додатки працюють незалежно, можуть знаходитися на окремих серверах або на одному сервері в різних компонентах. Дані користувачів, завдання, код користувачів зберігаються за допомогою СКБД PostgreSQL 11. Система складається з наступних компонентів.

1) Сервер авторизації - виконує функції реєстрація та авторизації користувачів. Для авторизації використовуються JWT токени, сама авторизація виконується згідно стандарту OAuth 2.0.

2) Сервер API - надає доступ до роботи з усіма даними всередині бази для клієнтських додатків. Для роботи з сервером користувачі передають свої токени. Розробникам додатків доступний REST-інтерфейс.

3) Сервер тестування – виконує компіляцію та тестування коду. Усі дані зберігаються тимчасово, після перезапуску видаляються. Підтримуються мови C, C++, Python та Pascal, додаткові мови можуть бути додані за допомогою зміни конфігурації. Одночасно може обробляти лише одне відправлення.

Створення децентралізованого додатку на базі платформи Ethereum

У процесі роботи створено додаток на базі децентралізованої мережі Ethereum [1], що дає змогу ознайомитися з технологіями та розглянути сферу використання блокчейн мережі, з урахуванням сучасних тенденцій розробки.

Був створений додаток, що дозволяє користувачам брати участь в лотереї, використовуючи технологію блокчейн, а саме розпоряджаючись власними токенами. Додаток був створений з використанням смарт-контрактів написаних на мові Solidity [1], статично типізованої мови, що працює на основі віртуальної машини Ethereum [2].

Solidity надає змогу розробникам програмного забезпечення створювати контракти, що містять бізнес-логіку, результуючу в транзакційні записи блокчейну.

Методологія розробки смарт-контрактів дозволяє дізнатися нові методи написання коду, тестування його роботи, та оцінити зручність оперування даними на децентралізованих мережах.

Ознайомлення з стандартом ERC-20 [3] дозволяє оцінити простоту розробки своєї криптовалюти на базі мережі Ethereum [2].

Зручність для взаємодії з користувачем, нам надає фреймворк на базі мови Javascript - Vue.js [4].

Колекція бібліотек web3.js [5] надає змогу взаємодіяти зі Ethereum нодами використовуючи HTTP або IPC з'єднання. Завдяки ній ми можемо працювати з децентралізованою базою даних, працювати з токенами, та взаємодіяти з іншими можливостями смарт контрактів, тощо.

Також використання розширення MetaMask [6] надає нам додатковий шар безпеки при підтвердженні транзакцій, та оперуванні балансом нашої чи будь-якої іншої криптовалюти.

Проект несе у собі тільки навчальний сенс, розглядаючи його сирці, можна зрозуміти методологію написання смарт контрактів, створення власних токенів, та методи взаємодії частини представлення з частиною бізнес-логіки у вигляді контрактів та блокчейн мережі.

Список літератури

1. Solidity. Solidity 0.5.3 documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://solidity.readthedocs.io/en/v0.5.3>
2. Ethereum Virtual Machine (EVM) Awesome List · ethereum/wiki Wiki [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: [https://github.com/ethereum/wiki/wiki/Ethereum-Virtual-Machine-\(EVM\)-Awesome-List](https://github.com/ethereum/wiki/wiki/Ethereum-Virtual-Machine-(EVM)-Awesome-List)
3. ERC20 standart [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://en.wikipedia.org/wiki/ERC-20>
4. Vue library [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://vuejs.org/>
5. web3.js library [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://web3js.readthedocs.io/en/1.0/>
6. MetaMask [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://metamask.io/>

Створення сайту для університетського центру кар'єри ЧНУ ім. Юрія Федьковича

Пошук роботи для студентів Чернівецького Національного Університету забирає великі проміжки часу. Після закінчення навчання, або ж під час літніх, або зимових канікул, студенти зазвичай планують знайти або постійне, або тимчасове місце роботи. Пошук роботи можна здійснювати самотужки, шукаючи в газетах, журналах, або ж на сайтах загального користування. В ЧНУ працює університетський центр кар'єри при профспілковій організації студентів. Працівники центру збирають інформацію про вакантні робочі місця на підприємствах, обробляють її, та подають її для студентів, які бажають знайти роботу. Студенти можуть звернутись до центру.

Уся ця інформація збирається лише у паперовому вигляді. Тобто усі процеси збирання, обробки та подачі інформації відбуваються виключно за допомогою людини, без використання технічних пристроїв.

З кожним роком данні про роботодавців та їхні вакансії зростають. Виникає проблема в швидкому доступі до інформації, її сортуванні, класифікації та зручному поданні для студентів. Рішенням даної проблеми є створення сайту, де кожен з студентів, який являється членом профспілкової організації студентів ЧНУ, може знайти потрібне йому місце роботи, подати заявку, що він бажає отримати певну роботу, а також отримати інформацію про цікаву роботу.

Головною причиною створення WEB додатку для університетського центру кар'єри стала трудність виконання головних операцій центру кар'єри виключно людьми.

Поставленою перед нами задачею було створення WEB додатку, який би міг виконувати наступні операції.

- Збирати інформацію від роботодавців, які виявили бажання подати вакансію.
- Збирати інформацію від студентів, які виявили бажання знайти певну роботу.

- Обробляти, сортувати, знаходити та вибирати з бази даних інформацію для студентів, щоб вони могли знайти потрібну для них роботу.

- Інформувати студентів, роботодавців та просто відвідувачів сайту. Публікувати інформацію про цікаві новинки у певних галузях роботи, задля стимулювання роботодавців запроваджувати у себе на робочих місцях нові технології та задля обізнаності студентів.

Доступ до сайту здійснюється за посиланням <http://stud-job.cv.ua>.

Ми вважаємо, що користуватися сайтом буде зручно не тільки студентам, які є членами профспілкової організації студентів ЧНУ, а й іншим користувачам, адже там можна не лише знайти роботу, а й почерпнути інформацію про те, як її правильно, швидко та якісно знайти. На сайті будуть цікаві поради про те, як писати резюме, правильно поводитися на співбесіді та інша корисна інформація, яка пригодиться усім. Це дозволяє підвищити рівень обізнаності студентів і стимулює роботодавців створювати комфортні умови для працівників.

У ході розробки WEB додатку, основною проблемою, з якою ми зіткнулися – реалізація гнучкого адміністрування, щоб зручний інтерфейс був не лише для звичайного користувача, а й для адміністраторів сайту (працівників центру кар'єри). Серверна частина додатку реалізована засобами PHP у поєднанні з бібліотекою AJAX.js, написаної мовою програмування JavaScript. Клієнтська частина – HTML, CSS, JavaScript.

Атоматизація документообігу навчального процесу

Microsoft Excel – одна з найзагадковіших і цікавих програм в пакеті MS Office. Цікава вона численними засобами автоматизації роботи, оформлення документів і обчислювальною потужністю. Загадковість ж її полягає в тому, що більшість користувачів застосовують у своїй роботі лише малу частину того, що може дати їм Excel. Це тим більше дивно, що спектр можливостей Excel майже безмежний: від створення добре оформлених бланків до програмування складних обчислень і моделювання різних процесів з візуалізацією параметрів у вигляді діаграм і графіків.

У роботі розглядається задача автоматизації підготовки документів для навчального процесу. Цей документ складається на початку кожного семестру для подачі методисту факультету, який складає розклад на семестр. Документ формується із урахуванням обсягу навантаження, що закріплено за кафедрою, та розподілу навантаження між викладачами. Враховуючи особливості навчального процесу, заняття з одного предмету проводяться різних типів та часто різними викладачами. Уся вхідна інформація зберігається у файлах програми Excel із визначеними іменами. Результуючий файл можна формувати як у програмі Excel, так і у програмі Word. Документ має табличну структуру.

Нами створено програму мовою VBA, вбудовану у Microsoft Excel, що виконує необхідні дії з навчальними документами. Запуск програми здійснюється через робочу область середовища VBA. Створена заздалегідь форма дозволяє спростити користування програмою. Користувач вказує шляхи до файлів із обсягом та розподілом та рядки з яких починаються відповідні дані. Після натискання кнопки “Сформувати дані до розкладу” програма підтягує наперед створені модулі (тобто коди) та

формує документ з даними до розкладу. Спочатку з обсягів програма витягує назви навчальних предметів та перевіряє їх на відповідність, а після цього з розподілу навчальних годин вона зчитує інформацію про викладачів, які читають дані предмети.

У ході розробки даних до розкладу основною проблемою була невідповідність назв предметів у файлі з обсягом та розподілом. Адже порівняння назв предметів здійснюється посимвольно. Паралельно із розробкою програми формувалися вимоги до вхідних файлів. Зокрема, варто нави предметів відображати однаково для скорочення часу формування результуючого документу.

Для використання даної розробки непотрібно додаткових ресурсів. Достатньо використати шаблонний документ обсягу та розподілу навантаження.

Для зручності користувача було розроблено простий інтерфейс на основі використання форм. Користувач може обрати вхідні файли, вказати інтервали аналізу вхідних файлів, зазначити ім'я та розташування вихідного файлу, зберегти сформований файл. Результуючий файл доступний для редагування.

Розроблений програмний засіб можна використовувати для будь-якого навчального підрозділу.

Розробка мобільного додатку для вивчення англійської мови

У наш час дуже складно знайти людину, яка б не користувався якимсь модним сучасним гаджетом. Хтось вважає, що гаджети – лише пуста трата часу в соціальних мережах, але це хибна думка. Сучасні пристрої надають нам дуже багато можливостей для розвитку. Існують різноманітні додатки, які підтримують пристрої. Зокрема, є розважальні та навчальні додатки. У даній роботі розроблено навчальний додаток для вивчення англійської мови.

Англійська мова – одна з найпопулярніших мов у світі. Кожна людина прагне вивчити цю мову чи то для подорожей, чи то для професії. Постає питання у швидкості та ефективності вивчення. В даній роботі створено додаток у середовищі Unity для вивчення англійської мови.

Unity – ігрова платформа для розробки ігор. За допомогою даної платформи можна розробляти ігри під операційні системи iOS, Android, Windows та інші. Редактор Unity має простий Drag & Drop інтерфейс, який легко налаштовувати. Він складається з різних вікон, завдяки чому можна проводити налагодження гри прямо в редакторі. Рушій підтримує дві сценарні мови: C#, JavaScript (модифікація). У даній роботі використовується мова C#, тому що вона має різноманітні бібліотеки та можливості, які дають змогу зробити додаток сучаснішим та зручнішим у використанні. Для запуску додатка на смартфоні з операційною системою Android необхідно скористатися засобом розробки Android SDK.

Android SDK – універсальний засіб розробки мобільних додатків для операційної системи Android. Він підтримує велику кількість мобільних пристроїв, серед яких виділяють: мобільні телефони, планшетні комп'ютери, розумні окуляри. На жаль, цей засіб не вбудований в ігрову платформу Unity, тому його треба встановлювати окремо для запуску на смартфоні.

Мобільний телефон – така річ, що завжди під рукою, тому зручно використовувати його для допомоги у вивченні, наприклад, англійської мови. Багато додатків спрямовані на те, щоб писати, вставляти слова в речення. Це звичайно допомагає при вивченні мови на рівні написання, проблема постає в спілкуванні, а саме правильна вимова, акценти та інше.

Даний додаток розроблений для того, щоб вивчити та мати базовий рівень у спілкуванні англійською мовою. Технологія така: озвучується слово, речення, далі за допомогою мікрофона (який є у всіх мобільних телефонах) треба повторити в точності те, що було сказано. Система перетворює звуки в текст, текст – у звук, який потім порівнюється щодо правильності вимови. Отже, телефон стає другом та співрозмовником. В залежності від правильності вимови отримуються бали. Кількість балів визначає успішність, за якою можна стежити. В даному додатку можна працювати зі своїм словником, повторювати вивчений матеріал, це надає можливість отримувати додаткові бали. Після проходження всіх рівнів розроблено проходження іспиту для закріплення знань.

Графічний інтерфейс додатка розроблений за допомогою інструментів Unity. Певні стилі розроблені програмою Adobe Photoshop, наприклад логотип програми, різні фони для додатка. Для об'єктів, які повторюються, створені колайдери для зручності і ефективності роботи. Інструкцію додатку записано на диктофоні, причому інструкція появляється лише тоді, коли ти зайшов перший раз у додаток, після введення імені. За допомогою мови C # створено скрипт без якого програма не може працювати. Скрипт включає в себе різноманітні бібліотеки, а найголовніше — включається бібліотека впізнання голосу. Слова та речення, які треба вимовити, зберігаються у файлах. За допомогою певних команд завантажуються в додаток. Весь навчальний текст узятий зі спеціальних перевірених книг.

Про поліноміальні і нарізно поліноміальні функції

Функція $f: E \rightarrow \mathbf{R}$, що задана на підмножині E числової прямої \mathbf{R} називається *поліноміальною*, якщо існує такий поліном

$$g(x) = a_0 + a_1x + \dots + a_nx^n,$$

що $f(x) = g(x)$ на E . Для підмножини E арифметичної площини \mathbf{R}^2 розглядають перерізи

$$E^x = \{v \in \mathbf{R}: (x, v) \in E\} \text{ і } E_y = \{u \in \mathbf{R}: (u, y) \in E\}.$$

Для функції $f: E \rightarrow \mathbf{R}$ ми розглядаємо її розрізи $f^x: E^x \rightarrow \mathbf{R}$ і $f_y: E_y \rightarrow \mathbf{R}$, що визначаються формулами $f^x(v) = f(x, v)$ і $f_y(u) = f(u, y)$. Функцію $f: E \rightarrow \mathbf{R}$ називають *нарізно поліноміальною*, якщо її розрізи f^x і f_y є поліноміальними для довільної точки $(x, y) \in E$. Сукупність усіх нарізно поліноміальних функцій $f: E \rightarrow \mathbf{R}$, позначають символом $S(E)$. Функцію $f: E \rightarrow \mathbf{R}$, називають *поліноміальною*, якщо існує такий многочлен

$$g(x, y) = \sum_{k,j=1}^n a_{kj} x^k y^j,$$

що $f(x, y) = g(x, y)$ на E . Сукупність поліноміальних функцій з $f: E \rightarrow \mathbf{R}$ позначають символом $P(E)$. Ясно, що $P(E) \subseteq S(E)$.

У праці [1] було показано, що у випадку $E = X \times Y$, де X, Y підмножини \mathbf{R} рівність $P(E) = S(E)$ виконується тоді і тільки тоді, коли серед множин X і Y щонайбільше одна зліченна. Але в загальному випадку відповідь на наступне питання невідоме.

Проблема 1. Які необхідні і достатні умови повинна задовольняти множина E , щоб для неї $P(E) = S(E)$?

Деякі достатні умови для рівності $P(E) = S(E)$ вказано у праці [2]. Тут ми досліджуємо випадок коли виконується

рівності $P(E) = S(E) = \mathbf{R}^E$ (нагадаємо, що Y^X – множина всіх відображень $f: X \rightarrow Y$).

Теорема 1. *Нехай $E \subseteq \mathbf{R}^2$. Для того, щоб $P(E) = S(E) = \mathbf{R}^E$ необхідно і досить, щоб множина E була скінченною.*

Список літератури

1. Косован В. М., Маслюченко В. К. Нарізно поліноміальні функції // Наук. вісник Чернівецького ун-ту. В. 374. Математика. –Чернівці: Рута, 2008. – С. 66 –74.
2. В.М Косован , В. К. Маслюченко Нарізно і сукупно поліноміальні функції на довільних підмножинах \mathbf{R}^n // Наук. вісник Чернівецького ун-ту. В. 454. Математика. – Чернівці: Рута, 2009. – С. 50 –53.

Структура фундаментального розв'язку задачі Коші для одного параболічного рівняння з операторами Бесселя різних порядків

Для одного параболічного рівняння 2-го порядку, що містить лінійну комбінацію операторів Бесселя різних порядків, знайдено у явному вигляді фундаментальний розв'язок задачі Коші та вивчено деякі його властивості.

Нехай $n \in \mathbb{N}$, $\mathbb{R}_+^n \equiv \{x = (x_1, \dots, x_n) \in \mathbb{R}^n \mid \forall l \in \{1, \dots, n\} : x_l > 0\}$.

Розглянемо задачу Коші

$$\partial_t u(t, x) = \sum_{l=1}^n A_l B_{x_l} u(t, x), \quad t > 0, \quad x \in \mathbb{R}_+^n, \quad (1)$$

$$u(t, x)|_{t=0} = \varphi(x), \quad x \in \mathbb{R}_+^n, \quad (2)$$

$$\partial_{x_l} u(t, x)|_{x_l=0} = 0, \quad t \geq 0, \quad x \in \mathbb{R}_+^n, \quad l \in \{1, \dots, n\}, \quad (3)$$

де $A_l > 0$ – сталі коефіцієнти, $B_{x_l} = \partial_{x_l}^2 + \frac{2\nu_l+1}{x_l} \partial_{x_l}$ – оператор Бесселя порядку $\nu_l > 0$ відносно змінної x_l , $l \in \{1, \dots, n\}$.

Методом перетворення Фур'є-Бесселя знайдено розв'язок задачі Коші (1) – (3) у вигляді інтеграла Пуассона

$$u(t, x) = \int_{\mathbb{R}_+^n} G(t, x; y) \varphi(y) \prod_{l=1}^n y_l^{2\nu_l+1} dy, \quad t > 0, \quad x \in \mathbb{R}_+^n.$$

Ядро G знайдено в явному вигляді

$$G(t, x; y) = \prod_{l=1}^n G_l(t, x_l; y_l),$$

де

$$G_l(t, x_l; y_l) = \frac{1}{2^{2\nu_l+1} A_l^{\nu_l+1} \Gamma(\nu_l+1) t^{\nu_l+1}} j_{\nu_l} \left(-\frac{x_l y_l}{2 A_l t} i \right) \times$$

$$\times \exp\left\{-\frac{x_l^2 + y_l^2}{4A_l t}\right\}, \quad t > 0, \quad \{x, y\} \subset \mathbb{R}_+^n,$$

– фундаментальний розв’язок задачі Коші для рівняння [1–3]
 $\partial_t u(t, x_l) = A_l B_{x_l}(t, x_l), \quad t > 0, x_l > 0, l \in \{1, \dots, n\}.$

Тут i – уявна одиниця, j_{ν_l} – функція Бесселя порядку ν_l .

Доведено, що G є фундаментальним розв’язком задачі Коші, та вивчено деякі властивості функції G .

Список літератури

1. Балабушенко Т. М., Івасишен С. Д., Лавренчук В. П., Мельничук Л. М. Фундаментальний розв’язок задачі Коші для деяких параболічних рівнянь з оператором Бесселя і зростаючими коефіцієнтами // Наук. вісник ЧНУ. Вип. 288. Математика: Зб. наук. праць. – Чернівці: Рута, 2006. – С. 5-11.

2. Мельничук Л. М. Структура та властивості фундаментального розв’язку задачі Коші для параболічного рівняння з операторами Бесселя // Буковинський математичний журнал. – Т. 4, №3-4.- Чернівці: Чернівецький нац. ун-т., 2016. – С. 119-122.

3. Скорохода В. Фундаментальний розв’язок задачі Коші для параболічного рівняння з операторами Бесселя // Матеріали студ. конф ЧНУ (25-27 квітня 2017 року). – Чернівці: Чернівецький нац. ун-т., 2017. – С. 55-56.

Інформаційна система “Допомога абітурієнту”

Світ навколо нас сповнений системами – механізмами, живими істотами, спільнотами людей, тобто системами механічними, біологічними, соціальними тощо.

У широкому розумінні інформаційною можна назвати будь-яку систему опрацювання інформації. Інформаційна система забезпечує приймання інформації, її перетворення, опрацювання, збереження і передачу результатів опрацювання споживачу: людині, машині, іншій інформаційній системі. Опрацювання інформації залежить від змісту вхідної інформації, але під час самого опрацювання інформація не осмислюється, а лише перетворюється згідно з попередньо розробленими алгоритмами.

У вузькому розумінні інформаційна система – це комплекс інформаційних, технічних, програмних та організаційних засобів, необхідних для автоматизованого опрацювання інформації [1].

Розробка інформаційної системи передбачає вирішення двох таких завдань:

- наповнення системи даними певної предметної галузі;
- створення (бажано графічного) інтерфейсу користувача для отримання необхідної інформації.

Зовнішнє незалежне оцінювання (ЗНО) – це форма іспитів для вступу до вищих навчальних закладів в Україні. ЗНО – комплекс організаційних процедур (передусім – тестування), що спрямований на визначення рівня навчальних досягнень випускників закладів загальної середньої освіти при їхньому вступі до закладів вищої освіти [2].

Для вступу до закладу вищої освіти на навчання за освітньо-кваліфікаційним рівнем бакалавра абітурієнт має подати сертифікати ЗНО з трьох конкурсних предметів. Конкурсні предмети, з яких вступник має надати сертифікати ЗНО, визначені Умовами прийому до закладів вищої освіти, в

залежності від спеціальності, обраної абітурієнтом. Для того щоб допомогти абітурієнту швидко і повно знайти необхідну інформацію про перелік необхідних сертифікатів ЗНО за обраною ним спеціальністю, нижні пороги балів, вагові коефіцієнти тощо, створюється система “ Допомога абітурієнту”. Ця система дає змогу абітурієнту:

- отримати необхідну інформацію, безпосередньо вибравши спеціальність за її назвою;
- вказавши факультет, передивитись інформацію про всі спеціальності, підготовка яких здійснюється на ньому;
- за переліком наявних сертифікатів ЗНО з'ясувати на які спеціальності та на які факультети можна подати документи.

При створенні інформаційної системи “Допомога абітурієнту” було обрано базу даних MySQL. Серверна частина додатку реалізована засобами мови Php, а клієнтська – за допомогою мови розмітки сторінки Html із використанням css-стилів.

Список літератури

1. Інформаційні системи, їх види [Електронний ресурс] – Режим доступу до ресурсу: <http://www.kievoit.ippo.kubg.edu.ua/kievoit/2013/95/95.html>.
2. Зовнішнє незалежне оцінювання [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Зовнішнє_незалежне_оцінювання.

Створення програмного забезпечення для плоттера

Метою роботи є створити програмне забезпечення для плоттера на мові програмування C++. Плоттер, або графобудівник – це пристрій для автоматичного креслення з великою точністю малюнків, схем та іншої графічної інформації. Конструкція плоттера - це дві рейки, які є осями Ox та Oy . Рух відбувається за допомогою крокових двигунів. На кінці однієї з рейок прикріплене перо, яке може опускатися та підніматися за допомогою серводвигуна [2].

Управління двигунами здійснює плата з мікроконтролером Arduino Uno. Arduino – це апаратно-програмні засоби для побудови простих схем автоматики та робототехніки [1]. Створення ПЗ для плоттера відбувалося за допомогою текстового редактора Atom із використанням розширення platformio-ide. Мовою програмування Arduino є C++[5].

Плоттер в даному проєкті призначений для побудови графіків математичних функцій. Точки для графіка можна згенерувати за допомогою iOS-додатка, де користувач вводить математичну формулу і вибирає точність. Згенеровані точки за допомогою http-запиту відправляються на сервер. Серверна програма написана на мові програмування Python.

Arduino за допомогою wi-fi модуля ESP8266 звертається до сервера із запитом, чи доступний масив точок. Стандартне ПЗ модуля використовує AT-команди, через які відбувається управління модулем: перевірка модуля, підключення до мережі, встановлення tcp-з'єднання з tcp-сервером, надсилання tcp-запиту і отримання відповіді від сервера, закриття з'єднання[3]. Передача даних із сервера відбувається пакетами по 50 байтів.

Оскільки пам'ять мікроконтролера дуже обмежена, масив прийнятих байт записується на карту пам'яті. Для цього в схему підключений модуль SD карти і використовується бібліотека SD для Arduino[4]. Після отримання всіх пакетів і запису їх на карту, починається управління двигунами. У циклі відбувається

зчитування потоку байт із карти пам'яті і конвертування його в структуру точки.

Для руху по осях використовуються двигуни 17HS1352-P4130. За один оберт двигун здійснює 200 кроків. Цю кількість можна збільшити за допомогою спеціальних перемичок на CNC shield. CNC shield – це спеціальна плата розширення для Arduino Uno. Вона може працювати з драйверами двигунів A4988, які використовуються в даному проєкті. CNC shield разом з драйверами дає можливість легко управляти двигунами. Для управління використовуються два виходи: один для задання напрямку руху, другий для генерації імпульсів для руху двигунів.

Структура програми містить 4 класи:

1. Esp8266 – клас для управління модулем ESP8266. Він містить методи для встановлення конфігурації модуля, надсилання AT-команд, надсилання запитів на сервер і отримання даних з сервера.
2. SDcard – клас для управління модулем SD карти. Він містить методи для ініціалізації, читання з файлу та запису у файл на карті пам'яті.
3. PointExtractor – клас для конвертування потоку байт в масив структур, які відповідають точкам.
4. Plotter – клас для управління двигунами. Містить методи для задання параметрів двигунів і руху від точки до точки.

Список літератури

1. Соммер У. Программирование микроконтроллерных плат Arduino/Freduino. – СПб.: БХВ-Петербург, 2012. - 256 с.
2. Петин В. Проекты с использованием контроллера Arduino. – СПб.: БХВ-Петербург, 2015. – 464 с.
3. Веб-сайт [електронний ресурс] // Режим доступу: <http://wiki.amperka.ru/продукты:esp8266-wifi-module>
4. Веб-сайт [електронний ресурс] // Режим доступу: <http://arduino-diy.com/arduino-SD-karta>
5. Веб-сайт [електронний ресурс] // Режим доступу: <https://all-arduino.ru/programmirovanie-arduino/>

Розробка сайту студмістечка Чернівецького Національного університету ім. Ю. Федьковича

Сучасною об'єктивною реальністю є широке впровадження у всі сфери життєдіяльності особи, суспільства та держави сучасних інформаційних технологій, розгортання на їх основі різноманітної складності локальних і глобальних інформаційних систем, призначених для прискорення обміну інформацією та доступу до різноманітних інформаційних джерел. Провівши аналіз сучасного розвитку інформаційних технологій і телекомунікацій, як один з критеріїв рівня розвитку держави та інструмент інтенсифікації всіх процесів у суспільстві, Україна, як і будь-яка інша країна, зацікавлена у розвитку інформаційної інфраструктури, в тому числі й шляхом використання у всіх сферах господарської діяльності можливостей, що надаються глобальними мережами передачі даних [1]. Всесвітня павутина являє собою надзвичайно перспективний засіб комунікації, що може пояснити його сучасний надзвичайний розвиток.

Саме тому, наший час великої популярності набувають так звані сайти-візитки окремих організацій, персон або підрозділів. Основним завданням таких сайтів є надання детальної інформації, у нашому випадку про місце навчання, в якому планується проведення тривалого часу. Перш ніж поступати в конкретний навчальний заклад, необхідно зібрати якомога більше інформації про заклад, а якщо ти не місцевий, то звичайно і про умови проживання у студентському містечку. Також зростає кількість людей, які воліють попередньо знайти всю необхідну інформацію в інтернеті. Охопити цей сегмент абітурієнтів звичайні вступні комісії не в змозі, тому що фізично розміщені на певних територіях[2]. Тому, розробка та впровадження веб-сайту є актуальною та доцільною.

Створений веб-ресурс дає можливість як абітурієнтам так і студентам бути в курсі всіх новин, що стосуються навчання, дозвілля та проживання.

Сайт складається з головної сторінки, сторінки з новинами, містить інформацію про проживання, поселення, дозвілля та контактну інформацію.

Вирішено розробити також можливість подання електронних даних для проживання в студентських гуртожитках, адже не завжди є можливість і час щоб занести заяву в деканат, а так все можна буде зробити в рази швидше, цьому також сприяє зручний і сучасний дизайн.

Для реалізації всього функціоналу використано мову програмування PHP, мову розмітки HTML та фреймворк Bootstrap. Для створення і роботи з базами даних використано MySQLAdmin.

Список літератури

1. Типи веб-сайтів [Електронний ресурс] – Режим доступу до ресурсу: <http://www.webtec.com.ua/uk/articles/index/view/2011-05-05/web-site>.
2. Сучасні інтернет-технології [Електронний ресурс] – Режим доступу: <https://sites.google.com/site/internettehnologii/com/sucasn-internet-tehnologii>.

Створення системи визначення динаміки стану річкових вод

Забруднення навколишнього середовища зараз досягло рівня, при якому спостерігається істотний його вплив на здоров'я населення. У басейни українських річок, за різними оцінками, щорічно скидається близько 10 млрд кубометрів стічних вод, у тому числі, чистих і нормативно очищених 6,5 млрд кубометрів і забруднених – 3,5 млрд кубометрів. Порушення норм якості води досягло рівнів, які ведуть до деградації водних екосистем, зниження продуктивності водойм [1].

У праці [2] сформульовано підхід до моделювання річкових процесів, який дозволяв би коректно враховувати всі стоки та витoki води в річковій системі. За цим підходом будь-яку ділянку річки пропонується розглядати як багатовимірну кібернетичну систему, на вхід якої надходять дані, які впливають на якість води у річці. Вихідною характеристикою системи є якість води в місцях водокористування.

Розглядається математична модель, яка описує зміну, наприклад, концентрації деякої хімічної речовини води у точках спостереження у часі

$$\ddot{x}(t) + V^* \dot{x}(t) - G_l^*(t) \cdot x(t) = 0, \quad (1.1)$$
$$x(0) = x_0, \dot{x}(0) = x_{00},$$

де

$$V^* = \frac{V}{D}, \quad K_l^* = \frac{k_l}{D}, \quad G_l^*(t) = \frac{G_l(t)}{D}, \quad l \in \{1, 2, 3\},$$
$$G_1^*(t) = -K_1^*, \quad (1.2)$$

$$G_2^*(t) = -K_2^*(1 - e^{-k_1 t}), \quad (1.3)$$

$$G_3^*(t) = -K_3^*(1 - e^{-k_2(1 - e^{-k_1 t})t}), \quad (1.4)$$

де x – це значення показника якості води, що моделюється; V – аналог швидкісного коефіцієнта (м/с); D – аналог турбулентнодифузного коефіцієнта (м²/с); T – інтервал часу, для якого виконується моделювання; x_0 – значення концентрації $x(t)$ у початковий момент часу $t = 0$;

$G_l(t)$ – функція, яка враховує вплив l -ї кількості груп послідовно одночасних процесів на зміну значення показника x ; k_l – коефіцієнт пропорційності між швидкістю протікання процесів l -ї групи та значенням показника, на який вони діють; множенням k_2 та k_3 на додаткові коефіцієнти враховуються впливи процесів попередніх груп на процеси наступної.

Інформаційна система дає змогу слідкувати за динамікою стану річкових вод у вибраних точках забруднень. При цьому, за допомогою математичної моделі (1.1) – (1.4), можна спрогнозувати час остаточного очищення води від певних видів забруднення. А також, у вищезазначеній системі, є можливість надання повідомлення про точки, в яких відбулось перевищення допустимих значень забруднення води.

Створено веб-сайт, back-частина якого була розроблена у середовищі Microsoft VisualStudio мовою програмування C#, з використанням програмної платформи .NET Framework, а front-частина – у середовищі Microsoft VisualStudioCode мовою програмування TypeScript, з використанням фреймворку Angular. Інформацію у базу даних Microsoft SQL Server можна вносити як через інтерфейс користувача, так і завантажувати з файлів .csv та .xml типу.

Список літератури

1. Стан водних ресурсів України [Електронний ресурс]. – Режим доступу: <http://gorizont-m.com.ua/ru/need-to-know/show/48>.
2. Мокін В.Б., Мокін Б.І. Математичні моделі та програми для оцінювання якості річкових вод. – Вінниця: Універсум-Вінниця, 2000. – 152 с.

Інформаційна система “Військкомат”

За ДСТУ 2392-94 інформаційна система (ІС) – комунікаційна система, що забезпечує збір, пошук, обробку та пересилку інформації [1]. У будь-якій інформаційній системі організовуються певні процеси, щоб виявити інформаційні потреби, здійснити відбір джерел інформації, збір інформації, виконати дії з її обробки, оцінити її значущість та подання у зручному вигляді, вивести інформацію для надання споживачам або для передачі в іншу систему. Основними факторами, які впливають на впровадження інформаційних систем, є потреби організацій та користувачів, а також наявність відповідних засобів для їх формування.

Причини, що спонукають організації впроваджувати інформаційні системи, з одного боку, зумовлюються прагненням збільшити продуктивність повсякденних робіт чи уникнути їх повторного проведення, а з іншого боку – бажанням підвищити ефективність управління діяльністю певної установи, приймаючи оптимальні та раціональні управлінські рішення [2]. Для цього часто здійснюють автоматизацію інформаційно-управлінської діяльності, до основних напрямів якої належать:

- автоматизація обробки документів із упровадженням систем для обробки тексту, автоматизація обміну інформацією через різноманітні види комунікацій [2];
- автоматизація діяльності менеджерів на базі комп’ютерних систем комплексних інформаційних систем, які надають допомогу в прийнятті рішень, та електронних секретарів, що переводять організацію праці менеджерів на якісно вищий щабель [2].

Поставленою перед нами задачею було створення інформаційної системи, яка би автоматизувала процес внесення

та обробки лікарських даних військовозобов'язаних Чернівецького обласного військкомату. З метою якісного призову громадян на строкову військову службу та прийняття громадян на військову службу за контрактом здійснюється медичний огляд для визначення ступеня придатності до військової служби. На основі результатів медичного огляду на військових лікарських комісіях (ВЛК) встановлюються остаточні рішення щодо кожного призовника. ІС користуватимуться як працівники військкоматів, так і лікарі ВЛК. ІС дозволяє працівникам військкоматів вести облік даних військовозобов'язаних, оформлювати їм направлення на проходження медичного огляду у ВЛК, отримувати дані поетапного проходження медичного огляду. Лікарі ВЛК мають змогу оформлювати огляди військовозобов'язаних згідно зі встановленими законодавством форматами і виносити остаточне рішення про їх придатність до військової служби. Крім того, можна отримати підсумкову інформацію у різних розрізах за отриманими даними роботи ВЛК за вибраний період.

У ході розробки інформаційної системи “Військкомат” основною проблемою, з якою ми зіткнулися, були обмежені технічні ресурси. Саме тому було обрано базу даних з відкритим вихідним кодом PostgreSQL, основними перевагами якої є надійність, невимогливість до обчислювальних ресурсів та можливість гнучкого адміністрування. Серверна частина додатку реалізована засобами мови C#, .Net Core, а клієнтська – за допомогою JavaScript бібліотек React.js та Redux.

Список літератури

1. ДСТУ 2392-94 Інформація та документація. Базові поняття. Терміни та визначення [Електронний ресурс] – Режим доступу до ресурсу: http://dbn.co.ua/load/normativy/dstu/dstu_2392_94/5-1-0-1166.
2. Необхідність інформаційних систем і технологій в управлінні підприємством [Електронний ресурс] – Режим доступу до ресурсу: http://ena.lp.edu.ua/bitstream/ntb/12500/1/62_327-332_Vis_722_menegment.pdf.

Властивості фундаментального розв'язку задачі Коші для одного параболічного рівняння з необмеженими коефіцієнтами і операторами Бесселя

Вивчено властивості знайденого в явному вигляді фундаментального розв'язку задачі Коші для одного параболічного рівняння другого порядку із зростаючими коефіцієнтами при похідних першого порядку та з операторами Бесселя.

Нехай $\mathbb{R}_+^5 \equiv \mathbb{R}^3 \times (0; +\infty) \times (0; +\infty)$, $x \equiv (x_1, x_2, x_3)$, $y \equiv (y_1, y_2)$, $u \equiv u(t, x, y)$.

Розглянемо задачу Коші

$$\partial_t u = \partial_{x_1}^2 u + \partial_{x_2}^2 u + \partial_{x_3}^2 u + \partial_{x_1}(x_1 u) + B_{y_1} u + B_{y_2} u, \quad t > 0, (x, y) \in \mathbb{R}_+^5, \quad (1)$$

$$u|_{t=0} = \varphi(x, y), (x, y) \in \mathbb{R}_+^5, \quad (2)$$

$$\partial_{y_j} u|_{y_j=0} = 0, \quad t > 0, x \in \mathbb{R}^3, j \in \{1; 2\}, \quad (3)$$

де $B_{y_j} \equiv \partial_{y_j}^2 + \frac{2\nu+1}{y_j} \partial_{y_j}$ – оператори Бесселя порядку $\nu \geq 0$ відносно змінної y_j , $j \in \{1, 2\}$.

Методом перетворення Фур'є-Бесселя знайдено розв'язок задачі Коші (1) – (3) у вигляді інтеграла Пуассона

$$u(t, x, y) = \int_{\mathbb{R}_+^5} G(t, x, y; 0, \xi, \eta) \varphi(\xi, \eta) \eta_1^{2\nu+1} \eta_2^{2\nu+1} d\xi d\eta,$$

$t > 0, (x, y) \in \mathbb{R}_+^5$.

Ядро G знайдено в явному вигляді і доведено, що воно є фундаментальним розв'язком задачі Коші (1) – (3).

Користуючись зображенням для G , доведено такі властивості.

Властивість 1.

$$\int_{\mathbb{R}_+^5} G(t, x, y; \tau, \beta, \eta) \eta_1^{2\nu+1} \eta_2^{2\nu+1} d\beta d\eta = e^{t-\tau}, \quad 0 \leq \tau < t,$$

тут $G(t, x, y; \tau, \beta, \eta)$ отримано з $G(t, x, y; 0, \beta, \eta)$ заміною t на $t - \tau$.

Властивість 2. Правильні оцінки похідних функції G

$$\begin{aligned} & \left| \partial_x^k \partial_y^m G(t, x, y; 0, \beta, \eta) \right| \leq \\ & \leq C_{km} t^{-2v-3-\frac{k_2+k_3+m_2+m_3}{2}} (1 - e^{-2t})^{-\frac{1+k_1+m_1}{2}} \times \\ & \times \exp \left\{ -m_1 t - c_1 \frac{(x_1 - \beta_1 e^{-t})^2}{1 - e^{-2t}} - c_2 \frac{(x_2 - \beta_2)^2}{t} - \right. \\ & \quad \left. - c_3 \frac{(x_3 - \beta_3)^2}{t} - c_4 \frac{(y_1 - \eta_1)^2}{t} - \right. \\ & \quad \left. - c_5 \frac{(y_2 - \eta_2)^2}{t} \right\} \cdot T_y^n \left[\exp \left\{ -\frac{\left(\frac{1}{4} - c_4\right) y_1^2 + \left(\frac{1}{4} - c_5\right) y_2^2}{t} \right\} \right], \\ & \quad t > 0, \{ (x, y), (\beta, \eta) \} \in \mathbb{R}_+^5, \end{aligned}$$

де $k \equiv (k_1, k_2, k_3)$, $m \equiv (m_1, m_2, m_3)$, $\{k_1, k_2, k_3, m_1, m_2, m_3\} \subset \mathbb{Z}_+$; $C_{km}, c_1, c_2, c_3, c_4, c_5$ – додатні сталі, $\{c_1, c_2, c_3\} \subset (0, 1)$, $\{c_4, c_5\} \subset \left(0; \frac{1}{4}\right)$, T_y^n – оператор узагальненого зсуву [1, 2].

Список літератури

1. Балабушенко Т.М., Івасишен С.Д., Лавренчук В.П., Мельничук Л.М. Фундаментальний розв'язок задачі Коші для деяких параболічних рівнянь з оператором Бесселя і зростаючими коефіцієнтами // Наук. вісник ЧНУ. Вип. 288. Математика: Зб. наук. праць. – Чернівці: Рута, 2006. – С. 5–11.
2. Мельничук Л.М. Структура та властивості фундаментального розв'язку задачі Коші для параболічного рівняння з операторами Бесселя // Буковинський математичний журнал. – Т. 4, № 3–4. – Чернівці: Чернівецький нац. ун-т, 2016. – С. 119 – 122.

Слабкі R -простори та їх застосування

Розглянемо топологічні простори X та Y і відображення $f: X \rightarrow Y$. Нагадаємо, що відображення f називається:

- *неперервним на множині X* , якщо для довільної точки $x \in X$ справедлива рівність $\lim_{y \rightarrow x} f(y) = f(x)$;
- *B_1 -відображенням або функцією першого класу Бера*, якщо існує послідовність неперервних функцій $f_n: X \rightarrow Y$, така, що $\lim_{n \rightarrow \infty} f_n(x) = f(x)$ для всіх $x \in X$.

Сукупність усіх функцій першого класу Бера між просторами X та Y ми позначаємо символом $B_1(X, Y)$.

Добре відомо [1, с. 431], що рівномірна границя $f: X \rightarrow \mathbb{R}$ послідовності неперервних функцій $f_n: X \rightarrow \mathbb{R}$ також є неперервною функцією, а рівномірна границя послідовності функцій $f_n: X \rightarrow \mathbb{R}$ першого класу Бера залишається функцією першого класу Бера [2].

Втім, не завжди рівномірна збіжність гарантує збереження тих чи інших властивостей дограничних функцій. Так, в [3] був наведений приклад підмножини $Y \subseteq \mathbb{R}^2$ і послідовності функцій $f_n: [0, 1] \rightarrow Y$ першого класу Бера, яка рівномірно збігається на відрізьку $[0, 1]$ до функції $f: [0, 1] \rightarrow Y$, що не належить до першого класу.

Отже, природно виникає питання про вивчення класів просторів значень Y , для яких система функцій між X та Y з певними властивостями замкнена відносно рівномірних границь.

Виявляється, що при дослідженні цього питання корисним є поняття так званого R -простору, введеного в [4]. Крім того, в [4] було доведено, що клас $B_1(X, Y)$ замкнений відносно взяття рівномірної границі, якщо $Y \in R$ -простором.

Оскільки клас R -просторів є сприятливим для розв'язання задачі про рівномірну границю, то цікавим є вивчення властивостей таких просторів.

Теорема 1. *Довільна опукла підмножина метричного топологічного векторного простору є R -простором.*

Теорема 2 [5]. *Одиничне коло*

$$S^1 = \{z \in \mathbb{C}: |z| = 1\}$$

з метрикою, індукованою з комплексної площини, є R -простором.

Виявляється, що інші замкнені криві (наприклад, еліпс) не є R -просторами навіть, якщо вони гомеоморфні до кола. Для того, щоб отримати твердження про рівномірні границі B_1 -функцій зі значеннями в таких просторах, нам довелося ввести поняття слабого R -простору.

Означення. Метричний простір (X, d) називається *слабким R -простором*, якщо існують послідовність $(\varepsilon_n)_{n=1}^{\infty}$ додатних чисел та послідовність $(R_n)_{n=1}^{\infty}$ функцій $R_n: X \times X \rightarrow X$, такі, що

- (1) ряд $\sum_{n=1}^{\infty} \varepsilon_n$ збіжний,
- (2) якщо $d(x, y) \leq \varepsilon_n$, то $R_n(x, y) = x$,
- (3) $d(R_n(x, y), y) \leq \varepsilon_n$

для довільних $x, y \in X$.

Теорема 3. Нехай X – топологічний простір, Y – слабкий R -простір. Тоді клас $B_1(X, Y)$ замкнений відносно рівномірних границь.

Список літератури

1. Фихтенгольц Г.М. Курс дифференциального и интегрального исчисления. Том 2, Москва: Наука, 1969.
2. Куратовский К. Топология. Том 1, Москва: Мир, 1966.
3. Карлова О., Михайлюк В. Функції першого класу Бера зі значеннями в метризовних просторах, Укр. мат. журн. **58** (4) (2006), 567-571.
4. Карлова О. Берівська класифікація відображень зі значеннями у підмножинах скінченновимірних просторів, Наук. вісн. Чернів. ун-ту. Вип. 239. Математика. (2005), 59-65.
5. Lukan M. R -spaces and uniform limits of sequences of functions, The 13th International Summer School in Analysis, Topology and Applications (July 29 – August 11, 2018, Vyshnytsya, Chernivtsi Region, Ukraine). Book of Abstracts. P. 24 – 25.

Використання цифрового ідентифікатора об'єкта в наукових публікаціях

Потужним інструментом у розповсюдженні наукових статей є цифровий ідентифікатор об'єктів (ЦІО) (англ. Digital Object Identifier, DOI) — серійний номер, який використовують для постійної та унікальної ідентифікації об'єктів будь-якого типу. Даний ідентифікатор підтримується та призначається об'єднанням видавців наукових публікацій Crossref, яке створене з метою розробки та підтримки всесвітньої високотехнологічної інфраструктури наукових комунікацій. Головним завданням об'єднання є сприяння широкому використанню інноваційних технологій та полегшення наукових досліджень. DOI дає змогу знайти документ навіть у випадку зміни його URL, яка може виникати зі зміною сайту, адже не залежить від розташування документа. Даний серійний номер потрібен для того, щоб ідентифікувати зразки інтелектуальної власності в цифровій мережі [1].

Для надання виданню DOI для поточного та архівного номерів журналу потрібно укласти відповідну угоду з ПА “Укрінформнаука” про співпрацю, оскільки у Національній академії наук України передплатне агентство “Укрінформнаука” є резидентом Crossref. Було підготовано клопотання для надання DOI Буковинському математичному журналу [2].

На сайті Crossref заповнено реєстраційну форму на Буковинський математичний журнал Т. 6, №1-2, 2018 рік. [3]

Кожна стаття для отримання DOI має містити метадані: правильний опис-посилання на цю статтю (скорочена англомовна назва журналу відповідно до ISSN, рік виходу, том, номер, сторінковий інтервал статті), прізвища авторів, назви установ, де вони працюють, назву статті, анотацію (резюме) до статті, ключові слова, пристатейні списки літератури, механізм переходу на повнотекстову версію статті (наприклад, у форматі *pdf*).

Додавання супровідних файлів надає можливість надати матеріали, що доповнюють статтю. Якість заповнення метаданих впливає на подальше просування статті у наукометричних базах.

Завдяки підготовленому клопотанню, заповненню реєстраційної форми та підготовці метаданих статей Буковинський математичний журнал: Т. 6, №1-2, 2018 рік видання має наявні DOI при своїх статтях: [4]

ON THE BORDER PROBLEM OF RING DOMAIN DEFORMATION

D. I. Anpilohov

Zaporizhzhia National Technical
University

N. V. Snizhko

Zaporizhzhia National Technical
University

DOI: <https://doi.org/10.31861/006>

Анотація

В даній роботі методом комплексних потенціалів Мусхелішвілі побудовано розв'язок граничної задачі теорії пружності для області в

БУКОВИНСЬКИЙ
МАТЕМАТИЧНИЙ
Ж У Р Н А Л

Том 6, № 1-2



Рис 1. Приклад статті з ідентифікатором DOI

Список літератури

1. Цифровий ідентифікатор об'єкта [Електронний ресурс] – Режим доступу: https://en.wikipedia.org/wiki/Digital_object_identifier.
2. Присвоєння цифрових ідентифікаторів DOI науковим публікаціям [Електронний ресурс] – Режим доступу: http://akademperiodyka.org.ua/sites/default/files/manuals/ukrInform_2_0.pdf.
3. Crossref [Електронний ресурс] – Режим доступу: <https://www.crossref.org>.
4. Буковинський математичний журнал [Електронний ресурс] – Режим доступу: <http://bmj.fmi.org.ua/index.php/adm/issue/view/40>.

Синхронізація каналів комп'ютерної мережі при передачі трафіку реального часу

При передачі по комп'ютерних мережах інформації, критичної до затримок, по декількох, у загальному випадку неоднорідних паралельних каналах, виникає проблема синхронізації цих каналів. Найбільш високі вимоги до якості передачі висуваються для мови, відео та команд керування.

Процес функціонування системи можна [2] відобразити GERT-мережею, якщо її поведінка розглядається через послідовні переходи з одного стану в інший C_1, C_2, \dots (число станів скінченне або нескінченне). Кожному з них приписана деяка ймовірність p_k ; ймовірності послідовності проходження станів визначається за правилом множення $P\{(C_1, \dots, C_n)\} = p_0 \dots p_n$. Кожній парі (C_j, C_k) відповідає умовна ймовірність p_{jk} ; якщо стан C_j досягнуто на певному кроці, то ймовірність переходу до стану C_k на наступному кроці дорівнює p_{jk} . Перехід системи зі стану в стан пов'язується з виконанням деякої операції, що описується випадковою величиною з відомим законом розподілу. У моделях GERT станам системи відповідають вузли графа, а виконуваним у системі операціям – дуги (гілки) графа. Випадкові величини, що приписані дугам GERT-мережі, повинні володіти властивістю адаптивності. Основними кроками при використанні GERT-мереж є:

- представлення системи у вигляді стохастичної мережі $G = (N, A)$, де N – множина GERT-вузлів, а A – множина дуг;
- визначення умовної ймовірності похідної функції моментів кожної дуги;
- обчислення W -функції кожної дуги;
- знаходження еквівалентної W -функції GERT-мережі $W_E(s)$, що виражає зв'язок еквівалентних W -функцій петель першого і r -х порядків. Петлею першого порядку є зв'язана послідовність орієнтованих гілок, кожен вузол яких є загальним для двох гілок. Петлею порядку r є множина не зв'язаних між собою петель першого порядку. Еквівалентна W -функція $W_E(s) = p_E M_E(s)$, де p_E – ймовірність виконання стоку, $M_E(s)$ – еквівалентна похідна функція моментів;

- знаходження математичного сподівання μ_{1E} і дисперсії $\sigma^2 = \mu_{2E} - (\mu_{1E})^2$ часу виконання GERT-мережі.

Аналіз процесів передачі пакетів на основі моделей GERT суттєво полегшується, якщо GERT-мережа розкладається на суму паралельних частинних графів. Іншими словами, еквівалентна W -функція GERT-мережі з декількома простими s - t -шляхами дорівнює сумі W -функцій частинних GERT-мереж, у кожній з яких міститься тільки один простий s - t -шлях.

Ми розглядаємо тільки GERT-мережі, для яких існують еквівалентний розклад на множину паралельно з'єднаних частинних графів $G_i^{(A)}$. Моделі такого роду можуть бути використані для аналізу і покращення характеристик типових протоколів передачі інформації.

Еквівалентна W -функція паралельно з'єднаних частинних графів $G_1^{(A)}, \dots, G_k^{(A)}$

$$W_E(s) = W_{G_1^{(A)}}(s) + \dots + W_{G_k^{(A)}}(s) \\ = \frac{\prod W_{A1}(s)}{1 - \sum \prod W_{A1}(s)} + \dots + \frac{\prod W_{Ak}(s)}{1 - \sum \prod W_{Ak}(s)}.$$

Дільник кожного доданку дорівнює добутку W -функцій гілок $W_{A1}(s)$, що складають i -й простий s - t -шлях. Знаменник кожного доданку дорівнює різниці між одиницею і сумою добутків W -функцій петель все можливих порядків $W_{Ai}(s)$, що не включають в себе ні початок, ні кінець GERT-мережі. В знаменнику петлі непарних порядків мають знак “мінус”, а парних – знак “плюс”.

Нами написана програма, яка дозволяє реалізувати вищепоставлену задачу.

Список літератури

1. Филлипс Д., Гарсиа-Диас А. Методы анализа сетей. – Москва: Мир, 1984. – 496 с.
2. Шибанов А.П., Кравчук Н.В., Башев А.В., Шибанова О.А. Синхронизация каналов компьютерной сети при передачи трафика реального времени // Вестник РГРТУ. – 2007, № 22. – С. 53 – 57.

Аналіз стохастичної моделі волатильності

Формула Блека-Шоулза та відкриття Чикагської опціонної біржі в квітні 1973 р. відкрили нову еру в фінансовій математиці. Формула Блека-Шоулза [1] широко використовується трейдерами для встановлення ціни звичайних опціонів при мінімальних ризиках. Проте особливо після ринкової кризи 1987 р. і появи потужних комп'ютерів та нових математичних технологій, багато зусиль віддано для створення нових моделей та для дослідження цінової інтерпретації. Зокрема, це видно з досліджень волатильності (показник, який характеризує тенденцію ринкової ціни або доходу змінюватися з часом), яка закладена в обмінах та не є сталою і проявляє себе через випадкові успіхи. Доведено, що короточасні грошові вкладення є ціннішими відносно інших вкладень, які передбачали Блек і Шоулз, і що виграшна ціна є статистично істотною та може змінюватися через деякий час.

В 1998 р. Гобсон і Роджерс конкретизували локальну волатильність за допомогою навантажених моментів минулих повернень. D_t -відхилення цін від тенденції, що визначається формулою

$$D_t = Z_t - \int_0^{+\infty} \lambda e^{-\lambda \tau} Z_{t-\tau} d\tau, \quad \lambda > 0 \quad (1)$$

де $Z_t = \log(e^{-rt} S_t)$. У (1) параметр λ дорівнює швидкості, з якою ціни змінюються. Гобсон і Роджерс припускають, що S_t є процесом, що задовольняє умову

$$dS_t = \mu(D_t) S_t dt + \sigma(D_t) S_t dB_t, \quad (2)$$

де μ і σ – детерміновані функції, а σ – додатне.

Природною основою для вивчення моделі Гобсона-Роджерса як з аналітичних, так і з числових точок зору, є рівняння Колмогорова [2]

$$a(\partial_{xx}u - \partial_x u) + x\partial_y u - \partial_t u = 0. \quad (3)$$

Вироджені рівняння типу Колмогорова природно виникають у задачі ціноутворення залежних від термінів вимог.

Розглядаються крайові задачі для рівняння (3), які розв'язуються числовим методом, запропонованим в [2]. Розглянено ціни опціонів з невеликою волатильністю та коротким терміном до погашення і у випадку високої волатильності і довшим терміном повернення. Отримані дані порівнемо із результатати застосування евклідового методу (з тими ж вхідними параметрами) і точним результатом.

Список літератури

1. Black F., Scholes M. The pricing of options and corporate liabilities // Journal of Political Economy, 1973, – **81**, № 3. – P. 637-659.
2. Di Francesco M., Foschi P., Pascucci A. Analysis of an uncertain volatility model // Journal of Applied Mathematics and Decision Sciences, 2006. – P. 1– 17.

Створення навчально - розважальної гри для дітей у середовищі Unity3D

Комп'ютери, смартфони, планшети, інтернет та соціальні мережі — усе це тепер є невід'ємною частиною нашого життя. Новітні технології дають можливість поліпшити процес навчання у школах та університетах. Навчання стало набагато простішим та цікавішим.

Звичайно, дітям більше хочеться гратися, ніж вчитися. Тож найкраще справляються із задачею розвиваючі ігри, які в ігровій формі можуть навчити дитину навіть чомусь складному, іноземній мові чи математиці. Для менших дітей гаджети можна використовувати у розвиваючих цілях, зокрема, для вивчення алфавіту, чисел чи кольорів.

Щоб зробити цікавішим процес вивчення математики в початковій школі, доцільно долучати до класичної методики різноманітні ігрові форми навчання. З такою метою була розроблена навчально-розважальна 2D гра в середовищі Unity3D.

Unity3D є сучасним кросплатформним середовищем для створення ігор і додатків, розробленим Unity Technologies. За допомогою даного середовища можна розробляти не тільки додатки для комп'ютерів, але і для мобільних пристроїв (наприклад, на базі Android), ігрових приставок і інших девайсів.

Редактор Unity має простий Drag & Drop інтерфейс, який легко налаштовувати; він складається з різних вікон, завдяки чому можна проводити налагодження гри прямо в редакторі. Рушій підтримує три сценарних мови: C#, JavaScript (модифікація).

Проект в Unity ділиться на сцени (рівні) — окремі файли, що містять свої ігрові світи зі своїм набором об'єктів, сценаріїв і налаштувань. Сцени можуть містити в собі як об'єкти (моделі), так і порожні ігрові об'єкти — ті, які не мають моделі. Об'єкти містять набори компонентів, з якими і взаємодіють скрипти. У об'єктів є назва, може бути тег (мітка) і шар, на якому він повинен відображатися.

Розроблена гра полягає у проходженні по місцевості, обходячи перешкоди та збираючи бонуси. Тут дитина вчиться швидко реагувати на небезпеки та приймати раціональні рішення. Для переходу на наступний рівень гри гравець повинен правильно розв'язати ряд прикладів.

Для організації розв'язання прикладів використовується база даних SQLite.

SQLite – одна з найбільш використовуваних систем управління базами даних. Головною перевагою SQLite є те, що для бази даних не потрібно сервера. База даних являє собою звичайний локальний файл, який ми можемо переміщати разом з головним файлом програми. Крім того, для запитів до бази даних можна використовувати стандартні вирази мови SQL.

При розробці гри використовувалися спрайти, скрипти, анімації. В грі присутні такі елементи, як дерева, вказівники, монстри, перешкоди, об'єкти, які потрібно збирати, та інші. На початку гри в гравця є 5 життів. Якщо в гравця закінчується життя, то його повертає на початок гри. Для того, щоб гравець міг виконувати якісь функції, були прописані скрипти мовою C#, прикріплені до нього.

Сучасні технології – це наше майбутнє. З кожним роком на нас чекає все більше й більше відкриттів та новинок. Для сучасних дітей та підлітків це вже є нормою життя.

Список літератури

1. Гаджети для освіти. URL: <https://learning.ua/blog/201612/hadzhety-dlia-osvity/>
2. Руководство Unity: Unity – Руководство. URL: <https://docs.unity3d.com/ru/current/Manual/index.html>
3. Unity. URL: <https://metanit.com/sharp/general.php>
4. Язык программирования C# и .NET. URL: <https://metanit.com/sharp/general.php>
5. SQLite Home Page. URL: <https://metanit.com/sharp/general.php>

Про рівносильність деяких згорткових співвідношень

При розв'язанні деякої задачі з теорії операторів у просторах послідовностей виникло питання про рівносильність певних згорткових співвідношень. У даному повідомленні вивчається це питання.

Нехай $X = \{x = (x_m)_{m=0}^\infty\}$ – деякий лінійний простір послідовностей комплексних чисел над полем \mathbb{C} .

Нехай для $m = 0, 1, 2, \dots$

$$e^{(m)} = (\underbrace{0, \dots, 0}_{m}, 1, 0, \dots).$$

Тоді

$$\forall x \in X: x = \sum_{m=0}^{\infty} x_m e^{(m)}.$$

Зафіксуємо послідовність ненульових комплексних чисел $(\alpha_m)_{m=0}^\infty$. Рівністю

$$I_\alpha x = \sum_{m=1}^{\infty} \frac{x_{m-1} \alpha_m}{\alpha_{m-1}} e^{(m)}$$

визначається лінійний неперервний оператор на X , який називається оператором узагальненого інтегрування, а формулою

$$x * y = \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} \frac{\alpha_0 \alpha_{n+m+1}}{\alpha_n \alpha_m} x_n y_m e^{(n+m+1)}, \quad x, y \in X, \quad (1)$$

Визначається нетривіальна згортка для оператора I_α на X , причому

$$I_\alpha x = e^{(0)} * x, \quad x \in X.$$

Розглянемо на X проектори P_0 та P_1 , які на довільну послідовність $x \in X$ діють відповідно за правилами

$$P_0 x = \sum_{m=0}^{\infty} x_{2m} e^{(2m)},$$

$$P_1 x = \sum_{m=0}^{\infty} x_{2m+1} e^{(2m+1)}.$$

Теорема. Нехай X – простір послідовностей, згортка $x * y$ визначається рівністю (1) для $\alpha_n = 1, n = 0, 1, \dots, a^{(0)}, a^{(1)}, b^{(0)}, b^{(1)}$ – деякі послідовності з X . Тоді рівносильними є такі згортчні співвідношення:

$$\begin{cases} b^{(0)} = a^{(0)} + (Ia^{(0)}) * (P_0 b^{(0)}) + a^{(1)} * (P_1 b^{(0)}), \\ b^{(1)} = a^{(1)} + (Ia^{(0)}) * (P_0 b^{(1)}) + a^{(1)} * (P_1 b^{(1)}), \end{cases}$$

та

$$\begin{cases} b^{(0)} = a^{(0)} + (Ib^{(0)}) * (P_0 a^{(0)}) + b^{(1)} * (P_1 a^{(0)}), \\ b^{(1)} = a^{(1)} + (Ib^{(0)}) * (P_0 a^{(1)}) + b^{(1)} * (P_1 a^{(1)}). \end{cases}$$

Використовуючи цю теорему, можна отримати рівносильність цих співвідношень і в загальному випадку.

Михайло Пастула
Наукові керівники – проф. Бігун Я.Й,
проф. Козьма Д.В. (Тираспольський
державний університет, Молдова)

Розв'язування задачі хижак-жертва з нелокальними умовами

Диференціальні рівняння, розв'язки яких задовольняють інтегральні умови, служать математичними моделями в біології, екології, при дослідженні оптичних явищ та в інших областях. Інтегральні умови можуть задаватись як на інтервалі існування розв'язку [1, 2], так і на його підмножині [3,4]. Побудувати розв'язок таких задач можна хіба що наближено.

У роботі [5] для рівняння з лінійно перетвореним аргументом й інтегральними умовами побудовано різницеву схему, модифіковано метод прогонки знаходження числового розв'язку та розроблено програму його знаходження.

У роботі розглядається система нелінійних рівнянь

$$\begin{cases} \dot{u} = (a - bv)u, \\ \dot{v} = (-c + du)v, \end{cases} \quad (1)$$

якою описується взаємодія популяцій хижака і жертви, з інтегральними умовами вигляду

$$u(\tau_1) = \int_{t_1}^{t_2} \alpha(t)u(t)dt \quad v(\tau_2) = \int_{t_3}^{t_4} \beta(t)v(t)dt, \quad (2)$$

де $\alpha(t)$ і $\beta(t)$ – задані неперервні функції, $0 \leq t_1 < t_2 \leq L, 0 \leq t_3 < t_4 \leq L, 0 \leq \tau_1 \leq \tau_2 \leq L$. Умови (2) включають як початкові, так і двоточкові крайові умови. Інтегральними складовими в (2) можна задати величину популяцій на відповідних часових інтервалах, якщо відома щільність їх розподілу.

Для задачі (1), (2) побудовано різницеву схему вигляду

$$\begin{cases} u_n - u_{n-1} = ah u_{n-1} - hb u_{n-1} v_{n-1}, \\ v_n - v_{n-1} = -ch v_{n-1} + hdu_{n-1} v_{n-1}. \quad N = \overline{1, N}. \end{cases}$$

Нехай $t_i = m_i h, i = \overline{1, 4}, \tau_j = p_j h, j = 1, 2, m_i, p_j \in N, h = L/N$. Тоді апроксимація умов (2) на підставі квадратурної формули трапецій набуває вигляду

$$h \sum_{i=m_1}^{m_2} \alpha_i u_i = q_1 u_{q_1} \quad h \sum_{i=m_3}^{m_4} \beta_i v_i = q_2 v_{q_2}.$$

Для одержаної системи нелінійних рівнянь наближений розв'язок будується ітераційним методом Ньютона. На кожній ітерації задача зводиться до розв'язування системи лінійних алгебраїчних рівнянь порядку $2(N+1)$ із розрідженою матрицею

$$\begin{aligned} (1 + bhu_{n-1}^{(k)})\varepsilon_n - (1 + ah)\varepsilon_{n-1} + bhu_{n-1}^{(k)}\delta_{n-1} &= \varphi_n, \\ (1 - dhv_{n-1}^{(k)})\varepsilon_n \delta_n - (1 - ch)\delta_{n-1} - dhv_{n-1}^{(k)}\varepsilon_{n-1} &= \psi_n, \\ h \sum_{i=m_1}^{m_2} \alpha_i \varepsilon_i + q_1 \varepsilon_{q_1} &= h \sum_{i=m_1}^{m_2} \alpha_i u_i^{(k)} + q_1 u_{q_1}, \\ h \sum_{i=m_3}^{m_4} \beta_i \delta_i + q_2 \delta_{q_2} &= h \sum_{i=m_3}^{m_4} \beta_i v_i^{(k)} + q_2 v_{q_2}, \end{aligned}$$

де

$$\begin{aligned} \varphi_n &= u_{n-1}^{(k)} - (1 - ah + bhu_n^{(k)})u_n^{(k)}, \\ \psi_n &= v_{n-1}^{(k)} - (1 + ch - dhv_n^{(k)})v_n^{(k)} \end{aligned}$$

На мові Python розроблено програму розв'язування цієї системи лінійних рівнянь та візуалізації результатів. Проведено тестування програми на модельних прикладах.

Список літератури

1. Benchohra M., Henderson J., Luka R., Ouahab A. Boundary Value Problems for Systems of Differential, Difference and Fractional Equation. – Kluwer: Dordrecht-Boston-London, Netherlands, 2016. – 307 p.
2. Jankowski T. Differential equations with integral boundary conditions -J. Comput. Appl. Math. – 2002, vol. 147. – P. 1–8.
3. Ahmad B., Ntouyas S.K. A study of higher-order nonlinear ordinary differential equations with four-point nonlocal integral boundary conditions - J. Appl. Math. Comput. – 2012, vol. 39. – P. 97–108.
4. Бігун Я.Й., Краснокутська І.В., Петришин Р.І. Усереднення в багаточастотних системах із лінійно перетвореними аргументами і точковими та інтегральними умовами – Буковинський матем. журнал. – 2016. – 4, № 3-4. – С. 30–35.
5. Pastula M., Bihun Ya, Cozma D. Difference approximation of differential equations with linearly transformed argument and nonlocal conditions - Materialele conferintelor științifice a studenților, Editia a LXVII-a – Coropcefnu Eduard. – Chisinau: S.n., 2018 (Tipografia UST). – P. 134–135.

Програмна реалізація емулятора машини з необмеженими регістрами засобами мови Java

Машина з необмеженими регістрами (МНР) – абстрактна обчислювальна машина [1], що містить безліч регістрів R_1, R_2, R_3, \dots , у кожен момент часу в яких зберігаються деякі цілі невід’ємні числа r_1, r_2, r_3, \dots відповідно.

МНР може змінювати вміст регістрів при виконанні команди. Скінченний пронумерований перелік команд утворює програму. Команди програми послідовно нумеруються числами 1, 2, 3,

МНР сприймає команди чотирьох типів [1]: команду занулення **Z**, команду додавання одиниці **S**, команду переадресації **T** і команду умовного переходу **J**.

МНР при виконанні заданої програми працює тактами. Для кожного такту послідовність чисел r_1, r_2, r_3, \dots у відповідних регістрах називається конфігурацією МНР.

Перший такт для початкової конфігурації – виконання команди з номером 1. Далі послідовно виконуватимуться команди з номерами 2, 3, і т. д. Цей природний послідовний порядок виконання команд триватиме до тих пір, поки не зустрінеться команда умовного переходу **J**, яка може його змінити.

Конфігурація МНР після її зупинки називається кінцевою, а відповідний вміст регістру R_1 , тобто число r_1 , вважається результатом роботи МНР.

Як і машина Тюрінга, МНР дозволяє здійснювати обчислення часткових числових функцій $f(x_1, x_2, \dots, x_n)$, аргументи і значення яких є цілими невід’ємними числами. При цьому початкова конфігурація формується наступним чином: $r_1 = x_1$,

$r_2 = x_2, \dots, r_n = x_n$, а значеннями решти регістрів є нулі. Після зупинки МНР вмістом регістру R_1 має бути значення $f(x_1, x_2, \dots, x_n)$. Якщо ж значення $f(x_1, x_2, \dots, x_n)$ було не визначене, то МНР повинна працювати беззупинно.

Нижче наведено приклад досить простої програми [1] для обчислення часткової числової функції $f(x, y) = x + y$:

1. J(2,3,5)
2. S(1)
3. S(3)
4. J(1,1,1)

Як показала практика, ефективні розробка, аналіз та тестування нетривіальних програм можливі лише при використанні якого-небудь емулятора МНР.

Враховуючи цей аспект, засобами мови програмування Java було успішно виконано реалізацію емулятора МНР.

Створений емулятор має дружній, інтуїтивно зрозумілий інтерфейс користувача та надає наступні можливості:

- 1) режим конструктора команд для формування коду програми МНР;
- 2) режим імпорту/експорту коду програми МНР;
- 3) засоби модифікації сформованого чи імпортованого коду програми МНР;
- 4) режим конструктора регістрів для задання потрібної початкової конфігурації МНР;
- 5) завантаження заданих початкової конфігурації та відповідного коду програми в пам'ять емулятора для подальшого виконання;
- 6) покрокове ручне чи повне автоматичне виконання програми МНР;
- 7) візуалізацію всіх відповідних конфігурацій МНР в процесі виконання програми;
- 8) підрахунок кількості виконаних команд програми;
- 9) демонстрацію результату роботи МНР.

Список літератури

1. Крупский В.Н., Плиско В.Е. Теория алгоритмов. – М.: Издательский центр "Академия", 2009. – 208 с.

Формування в учнів навичок доведень математичних тверджень при вивченні геометрії

Навчання доведенням теорем відіграє важливу роль в розвитку дедуктивно-математичного мислення і загальних здібностей учнів. Вже з першого класу необхідно прививати учням навички в аргументуванні та обґрунтуванні. Це сприяло б підготовці їх до здійснення доведень в середніх і старших класах і виявилось б корисним надалі в самостійному житті, де також приходиться логічно мислити, обґрунтовувати, доводити. Крім того, вміння доводити відповідає потребам навчальної діяльності самого учня. Тому успіх його навчання тісно пов'язаний з навиками в самостійному проведенні доведень.

Проте часто в школі більше звертають увагу на здобуття учнями конкретних знань, умінь (так як їх легко виявити, перевірити). Розвитку ж здібностей, які проявляться пізніше, через кілька років, не завжди приділяють достатньо уваги. У результаті багато учнів або зовсім не мають навиків в доведенні теорем, або ж ці навички явно недостатні.

Як же навчити учнів самостійно проводити доведення? Класичне правило говорить: тільки шляхом розгляду великої кількості доведень теорем. Це вірно. Але вірно й те, що в підручниках з математики міститься дуже велика кількість доведених теорем, а практика показує, що небагато учнів можуть проводити ці доведення самостійно. Значить, самого тільки розгляду багатьох доведень ще не досить для того, щоб навчити учнів доводити.

Дуже велику роль в розвитку навиків самостійного проведення учнями доведень теорем відіграє мотивація: учень повинен побачити необхідність здійснення доведення. Тому, зокрема, істинність доводжуваної теореми не повинна бути самоочевидною.

Передова педагогічна думка працює зараз над ліквідацією розриву між колосальними зусиллями педагогів і антипатією учнів до доведень. І найбільш благодатним ґрунтом для досягнення цієї мети є геометрія, яка більше ніж інші шкільні предмети будується на дедуктивні логічній основі.

Один з можливих варіантів дедуктивного викладання курсу геометрії поданий в навчальному посібнику О. В. Погорелова [1, 2]. Автор дотримується позиції збереження дедуктивного методу побудови геометрії, збереження віками досягнутої ідеї доведення всіх її теорем, завдяки якій навчання найбільш природно поєднується з розвитком мислення в учнів і виховання в них матеріалістичного світогляду.

Навчальний посібник О. В. Погорелова являє собою систематичний виклад геометричного матеріалу, що ґрунтується на простій системі аксіом. Автор вважає, що шкільний курс геометрії повинен бути аксіоматичним від самого початку вивчення. Тільки при такій побудові посібника можна досягти найважливішої мети: пізнання учнями властивостей абстрактних просторових форм навколишнього світу і навчання їх логічно міркувати, аргументувати, доводити. При цьому не повинна ставитись мета – навчання аксіоматичним доведенням, а поступове, глибоке оволодіння учнями ідеєю логічної впорядкованості геометричних фактів, їхнього наукового взаємозв'язку. На аксіоматичній основі в посібнику додаються нові твердження, які після доведення одержують статус теорем. Частина тверджень, які раніше в підручниках вводились як теореми, подається у вигляді задач на доведення (так званих задач-теорем).

Прийняття підходу «доводити все» створює потужну цільову психологічну настанову учням, що особливо важливо на перших порах навчання геометрії. Тому саме на цьому етапі необхідна скрупульозність в прослідковуванні логічних зв'язків.

Список літератури

1. Погорелов О. В. Геометрія. Планіметрія: Підручник для 7-9 класів середньої школи. – К: Освіта, 1998. – 223с.
2. Погорелов О. В. Геометрія. Стереометрія: Підручник для 10-11 класів середньої школи. – К: Освіта, 2001. – 128с.

Децентралізований додаток на основі блокчейн

В епоху цифрових технологій децентралізованість даних грає велику роль в захисті інформації від підробок, її цілісності. Саме тут у гру вступає блокчейн, технологія, що набула великої популярності в останні роки. Воно й не дивно, адже завдяки ньому ми можемо безпечно зберігати свої дані, не переживаючи за їх цілісність, розпоряджатись своїми фінансами, виконувати важкі обчислення на декількох машинах одночасно й багато іншого.

В процесі розробки був створений додаток, що дозволяє користувачам проводити аукціон з продажу майна, використовуючи технологію блокчейн, а саме розпоряджаючись власними токенами. Додаток був створений з використанням смарт-контрактів написаних на мові Solidity [1], статично типізованої мови, що працює на основі віртуальної машини Ethereum (EVM) [2]. Мова була запропонована в серпні 2014 року Гейвіном Вудом. Solidity надає змогу розробникам програмного забезпечення створювати контракти, що містять бізнес-логіку, результуючу в транзакційні записи блокчейну.

Додаток використовує токен, створений за стандартом ERC20 [3], що дозволяє нам його пересилати з одного крипто-гаманця на інший, надавати дозвіл іншим користувачам користуватись нашими токенами, тощо. Для захищеної, зручної взаємодії з токеном використовував розширення MetaMask.

Для відображення інформації було використано javascript фреймворк Vue.js [4]. Vue використовує синтаксис шаблонів на основі HTML, що дозволяє декларативно зв'язувати рендеринг DOM з основними екземплярами даних в Vue. Всі Vue шаблони валідні HTML, і можуть бути розпарсені браузером та HTML парсерами. В середині Vue компілює шаблони в рендерингові функції віртуального DOM. В поєднанні з реактивною системою,

Vue здатний розумно обчислити кількість компонентів для рендерингу та застосувати мінімальну кількість маніпуляцій з DOM, коли стан застосунку зміниться. Для стилізації компонентів на сторінці використовував Vue-Bootstrap - бібліотеку яка надає можливість, швидко й просто користуватись готовими, стилізованими візуальними об'єктами.

Колекція бібліотек web3.js надає змогу взаємодіяти зі Ethereum нодами використовуючи HTTP або IPC з'єднання. Завдяки ній ми можемо отримувати інформацію з децентралізованої бази даних, розпоряджатись власними токенами, надсилати інформацію на ноди, тощо.

Проект був створений для знайомства з роботою ефірівських нод, смарт-контрактів, особливостей роботи з ERC20 токенами. Значна увага приділялась використанню фреймворку Vue.js, його взаємодії з бібліотекою web3.js для використання власних створених токенів.

Список літератури

1. Solidity. Solidity 0.5.3 documentation [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://solidity.readthedocs.io/en/v0.5.3>
2. Ethereum Virtual Machine (EVM) Awesome List · ethereum/wiki Wiki [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: [https://github.com/ethereum/wiki/wiki/Ethereum-Virtual-Machine-\(EVM\)-Awesome-List](https://github.com/ethereum/wiki/wiki/Ethereum-Virtual-Machine-(EVM)-Awesome-List)
3. Etherscan Token Tracker [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://etherscan.io/tokens>
4. Introduction — Vue.js [Електронний ресурс]: [Веб-сайт]. — Електронні дані. — Режим доступу: <https://vuejs.org/v2/guide/>

Розробка ілюстративних матеріалів для вивчення розділу програмування “Бази даних”

Бази даних – це інструмент, за допомогою якого можна збирати та упорядковувати інформацію. У базах даних можна зберігати найрізноманітнішу інформацію (про товари, клієнтів, постачальників, покупців, робітників і т.п.). Ці дані можуть бути у довільному форматі: числовому, текстовому, дати і часу, аудіо, відео. Більшість цієї інформації можна зберегти у текстових файлах, або у таблицях наприклад Excel або у Word.

Бази даних легко вирішують наступні проблеми.

1. Збереження довільного об'єму інформації.
2. Легкість доступу до інформації.
3. Легкість її зміни.
4. Точність і достовірність.
5. Повнота інформації.
6. Безпека.
7. Відсутність дублювання.

Розглядають різні структури для збереження даних. Популярною є реляційну модель представлення даних. Для її розробки використовують основи теорії реляційних баз даних. Ця теорія базується на положеннях реляційної алгебри.

Реляційна алгебра була введена Е. Коддом при розробці реляційної моделі. Реляційна алгебра складається з набору операторів, які використовують відношення в якості операндів і повертають відношення, як результат. Реляційна алгебра включає вісім операцій:

- 1) традиційні операції з множинами - об'єднання, перетин, різниця і декартовий добуток;
- 2) спеціальні реляційні операції - вибірка, проекція, селекція і ділення.

В роботі запропоновано візуалізацію реляційних операцій (об'єднання, перетин, різниця, проекція і селекція) для кращого

розуміння студентами основ реляційної алгебри. Для ілюстрації роботи було використано технологію DirectX.

DirectX – це набір API функцій (Application Programming Interface, тобто це є набір значень підпрограм, засобів для створення програмного забезпечення, це також набір чітко визначених методів для взаємодії різних компонентів. API надає програмісту всі можливості для розробки програмного забезпечення). Ці функції розроблені для простого та ефективного вирішення завдань, особливо пов'язаних з написанням відеоігор під Microsoft Windows.

DirectX - це ніби сполучна ланка, яка перетворює команди програми в зрозумілі команди для пристроїв. Завдяки технології DirectX, розвиток комп'ютерних технологій стрімко почала зростати в геометричній прогресії. Нові технології відкривають широкі можливості по створенню нових додатків і найрізноманітніших програм. За це можна подякувати корпорацію Microsoft.

Отже, засобами DirectX були розроблені відео ролики для демонстрації роботи основних операцій реляційної алгебри. Ці розробки будуть використовуватися у вивченні розділу "Бази даних".

Побудова iOS-додатку для побудови графіків і комунікації з плоттером

Метою роботи є створення мобільного додатку для операційної системи iOS мовою програмування Swift, який відображає графіки простих функцій, реалізує відправлення точок на сервер для комунікації додатку із плоттером.

Щоб побудувати та графічно відобразити графік функції, користувачу додатку необхідно відкрити сторінку, де у текстовому полі потрібно ввести функцію. Для зручності введення математичної формули була розроблена клавіатура, яка містить цифри, деякі тригонометричні функції, арифметичні операції та ін. Після додавання функції, список всіх графіків можна побачити у спеціальній таблиці, де їх можна видаляти чи редагувати.

Всі графіки відображаються у спеціальній графічній зоні на основній сторінці. Графічна зона містить сітку із позначками координат, яка динамічно підлаштовується під конкретний масштаб. Щоб масштабувати графік, потрібно використовувати відповідний жест двома пальцями. Згодом можна відправити на сервер, та намалювати плоттером лише ті точки, які входять у масштабовану зону.

Графіки функцій будуються шляхом з'єднання масиву точок, які обчислюються із математичних формул, заданих користувачем. Масив точок обраховується в циклі із заданим кроком та межами. Межі - це межі графіку по осі X (два дробових числа from та to), а значення кроку може задати користувач за допомогою слайдеру, який розміщений на головній сторінці. За замовчуванням значення слайдеру = 0.1.

У додатку є можливість отримати точки із файлу з розширенням .gcode [1]. Gcode-файл містить прості команди, для управління ЧПУ станками. Після зчитування інформації із файлу користувач має можливість конвертувати її у структуру, яка відповідає спеціально узгодженому протоколу.

Отриманий масив точок можна надіслати на сервер у форматі json, використовуючи http запит [2]. Для економії пам'яті та інтернет трафіку слайдером можна регулювати точність

побудови графіку (при заданій високій точності обчислюється велика кількість точок).

Серверна програма написана на мові програмування Python. Задача HTTP-сервера полягає в тому, щоб приймати POST-запити від iOS-додатку. Для створення HTTP-сервера використовується стандартний модуль `http.server`, який містить класи `HTTPServer` та `BaseHTTPRequestHandler`. Для створення сервера вказуємо хост і порт. Клас `BaseHTTPRequestHandler` використовується як базовий для реалізації власного обробника запитів. Клас `RequestHandler` є наслідником класу `BaseHTTPRequestHandler` і має реалізацію для метода `do_POST()`.

Частиною програми є TCP-сервер, який використовується для зв'язку із wi-fi модулем плоттера. Для створення TCP-сервера використовується стандартний модуль `socket`. В методі `runTCPserver()` відбувається створення нового сокета, його зв'язування з ip-адресою і портом, прослуховування порта і очікування підключення, отримання даних після підключення і надсилання відповіді. Wi-fi модуль підключається до TCP-сервера і надсилає повідомлення «GET». Якщо на сервері в цей час є доступний масив точок, TCP-сервер надсилає його відповідь модулю. Точки передаються потоком байтів. Передача всього масиву здійснюється за декілька запитів, оскільки у відповідь на один запит сервер надсилає пакет довжиною 50 байтів, що дорівнює 10-ти точкам. Якщо при підключенні модуля до сервера даних не було, сервер відсилає відповідь «NO DATA».

HTTP-сервер і TCP-сервер запускаються в двох потоках. Для цього використовується модуль `threading`.

Список літератури

1. Веб-сайт [електронний ресурс] // Режим доступу: <https://3dtoday.ru/blogs/xedos/gcode-the-basics/>
2. Веб-сайт [електронний ресурс] // Режим доступу: <https://habr.com/ru/post/330760/>

Дослідження неперервних константиноїдів

Якщо розглянути дві функції $f(x)$ та $g(x)$, хоча б одна з яких є сталою, то, очевидно, що тоді хоча б одна з композицій $f(g(x))$ чи $g(f(x))$ буде сталою.

Виникає природне питання: чи можна для не сталої функції $f(x)$ підібрати таку не сталу функцію $g(x)$, що

$$f(g(x)) = g(f(x)) = 0$$

для всіх $x \in \mathbb{R}$? Звісно, в правій частині останньої рівності можна розглядати не тільки нуль, а й інші константи.

Виявляється, що відповідь на це питання позитивна. Розглянемо функції $f(x) = [x]$ та $g(x) = \{x\}$. Зрозуміло, що вони не сталі, але

$$[\{x\}] = \{[x]\} = 0. \quad (1)$$

З іншого боку, якщо розглянути функцію $f(x) = x^2$, то для неї ми вже не зможемо підібрати не сталу функцію $g(x)$, яка задовольняє рівності (1).

Таким чином, цікавою задачею є знайти необхідні і достатні умови на функцію $f(x)$ (з певних класів: наприклад, монотонних, неперервних, диференційовних функцій, тощо) для існування не сталої функції $g(x)$, яка задовольняє рівності (1).

Функцію f назовемо *константиноїдом*, якщо для неї існує функція g така, що

$$f \circ g = g \circ f = C,$$

де $C \in \mathbb{R}$.

Функція f називається *власним константиноїдом n -го степеня*, якщо

$$\underbrace{f \circ f \circ \dots \circ f}_{n \text{ разів}} = C.$$

Вдалося отримати такі результати.

Теорема 1. Функція є константиною тоді і тільки тоді, коли

(1) $E_f \neq \mathbf{R}$,

(2) існують такі різні числа $a, b \in \mathbf{R}$, що $f(a) = f(b) = a$.

Зауваження. Для неперервних f і g достатньо додати, щоб множина A була проміжком.

Теорема 2. Функція є власним константиною тоді і тільки тоді коли її можна зобразити в наступному вигляді:

$$f = \begin{cases} k_i, & x \in A_i \\ C, & x \in A, \end{cases}$$

де k_i – функції, A_i, A – числові множини, $i \in \{1, \dots, n-1\}$, з наступними умовами:

1) область визначення функції f – множина всіх дійсних чисел, тобто,

$$\bigcup_{i=1}^n A_i \cup A = \mathbf{R};$$

2) область значень i -ої функції є підмножиною A_{i+1} (крім $i = n-1$);

3) область значень $(n-1)$ -ої функції та константа C міститься в A ;

4) область значень функції f та множина A_1 не перетинаються.

Мотивація навчання на уроках математики

Сучасна школа переорієнтовується на визнання особистості дитини найвищою цінністю. Особистість стає цінністю суспільства, а метою освіти – розвиток особистості. Для розвитку особистості математика має чималі можливості. Тому завданням вчителів математики – реально забезпечити якісне навчання, цілеспрямовано готувати школярів до глибокого і свідомого вивчення та розуміння математики в об’ємі шкільного курсу, створювати можливості для виявлення здібних і обдарованих дітей, розвивати інтерес до науково-дослідницької роботи. Весь процес навчання “математиці” має невичерпні можливості для розвитку людини, а особистісно зорієнтоване навчання забезпечує становлення системи особистісних освітніх цілей учня [1].

Мотивація, або прагнення дитини до навчання, є одним з найважливіших чинників, що забезпечують успішне сприйняття і засвоєння учнями програмового матеріалу.

Мотивація є або внутрішньою, або зовнішньою, залежно від ставлення дитини до процесу отримання знань. Ретельна діагностика навчальної діяльності кожного конкретного учня дає можливість педагогу створити сприятливі умови для формування і закріплення позитивної мотивації навчання. Мотиваційна сфера формується насамперед самою особистістю і важко піддається змінам, впливу ззовні.

Одна з умов зовнішньої мотивації – це впевненість особистості в наявності безпосереднього зв’язку між здійснюваною поведінкою та її наслідками, а основними елементами її є зовнішні стимули.

Ознаками зовнішньої мотивації навчання є: навчання заради навчання, без задоволення від діяльності або без зацікавленості; для особистої вигоди; заради успіху або уникнення невдач; з примусу або під тиском; навчання, що ґрунтується на моральних обов’язках та на соціальних цілях, вимогах і цінностях.

Учні із зовнішньою мотивацією, як правило, не отримують задоволення від подолання труднощів під час розв’язування

задач. Вони вибирають найпростіші завдання і розв'язують лише те, що необхідно для отримання певної оцінки.

Цей тип мотивації не тривалий у часі, поверхневий і на уроках математики його краще застосовувати лише в окремих випадках, коли: колектив класу має невисоку або різнорівневу підготовку; матеріал для вивчення складний.

Внутрішня мотивація, має як правило, стійкий характер і безпосередньо міститься в матеріалі, що вивчається. Для посилення внутрішньої мотивації учитель мусить певним чином опрацювати матеріал, зробити цікавим, взаємопов'язаним, орієнтованим на досягнення певних навчальних цілей і засвоєння конкретних дій. Для цього навчальний матеріал потрібно структурувати, він має бути доступним і зрозумілим для учня, а його засвоєння – свідомим.

Учні з внутрішньою мотивацією навчання демонструють інтелектуальну гнучкість і фантазію, а предмет, що вивчається, притягує їх ще й емоційно. Вони розв'яжуть найскладнішу задачу, прагнучи не просто засвоїти матеріал, а й зрозуміти його. Абсолютна внутрішня мотивація виникає в тому випадку, коли головною метою учня є отримання задоволення від самого навчання [2].

При підготовці до уроку математики, має бути ретельно продумана мотивація на рівні міжпредметних зв'язків, визначена значимість теми уроку для розвитку науки, повсякденного життя, для розв'язання економічних проблем, для пізнання світу, фактів та явищ підвищує усвідомлення матеріалу, що вивчається.

Список літератури

1. Андрух Ю.О. Методичні рекомендації по використанню актуальних питань мотивації навчання на уроках математики. – Чернівці, 2009. – 36 с.
2. Маркова А.К. Формирование мотивации учения в школьном возрасте. – М.: Просвещение, 1983. – 100 с.

Надія Штирбу
Науковий керівник – проф. Пукальський І.Д.

Багатоточкова крайова задача для параболічного рівняння з нерівностями

В області $Q = [t_0, t_{N+1}] \times D$ розглянуто задачу знаходження функції $u(t, x)$, яка при $(t, x) \in Q \setminus (\bigcup_{k=0}^N (Q \cap (t = t_k)))$ задовольняє рівняння

$$\begin{aligned} \partial_t u - \sum_{i,j=1}^n a_{i,j}(t, x) \partial_{x_i} \partial_{x_j} u + \sum_{i=1}^n a_i(t, x) \partial_{x_i} u + \\ + a_0(t, x) u = f(t, x), \end{aligned} \quad (1)$$

багатоточкові умови за часовою змінною

$$u(t_k + 0, x) = \varphi_k(x), \quad x \in Q \cap (t = t_k), \quad k \in \{0, 1, \dots, N\}, \quad (2)$$

і крайові умови

$$\begin{aligned} Bu|_{\Gamma} = \left(\sum_{j=1}^n b_j(t, x) \partial_{x_j} u + b_0(t, x) u \right) |_{\Gamma} \geq g(t, x), \\ u|_{\Gamma} \geq 0, \quad u(Bu - g)|_{\Gamma}, \end{aligned} \quad (3)$$

де $\Gamma = [t_0, t_{N+1}] \times \partial D$, ∂D – межа області D , $0 \leq t_0 < \dots < t_{N+1}$.

Дослідження задачі (1)–(3) проведено за таких обмежень:

а) рівняння (1) рівномірно параболічне [1], коефіцієнти

$$a_{i,j} \in C^a(Q), \quad a_i \in C^a(Q), \quad a_0 \in C^a(Q), \quad a_0 > 0;$$

б) межа області D належить класу C^{2+a} , функція $f \in C^a(Q)$,

$$\varphi_k \in C^{2+a}(Q \cap (t = t_k)), \quad g \in C^{1+a}(\Gamma_k), \quad b_j \in C^{1+a}(\Gamma_k)$$

$$b_0 \in C^{1+a}(\Gamma_k), \quad b_0|_{\Gamma} > 0, \quad (\varphi_k(B\varphi_k - g))|_{\Gamma \cap (t=t_k)} = 0, \quad \text{де}$$

$$\Gamma_k = [t_k, t_{k+1}] \times \partial D, \quad k \in \{0, 1, \dots, N\}.$$

Правильні наступні теореми.

Теорема 1. Нехай u – класичний розв’язок задачі (1)–(3) в області $Q_k = [t_k, t_{k+1}] \times D$ і виконані умови а) та б). Тоді в Q_k правильна оцінка

$$\|u\|_{C(Q_k)} \leq \|\varphi_k\|_{C(Q_k \cap (t=t_k))} + \|fa_0^{-1}\|_{C(Q_k)} + \|g\|_{C(\Gamma_k)}.$$

Теорема 2. Якщо для задачі (1)–(3) виконані умови а), б), то в областях Q_k існує єдиний розв’язок задачі (1)–(3) і правильна нерівність

$$\|u\|_{C^{2+a}(Q_k)} \leq C(\|\varphi_k\|_{C^{2+a}(Q_k \cap (t=t_k))} + \|f\|_{C^a(Q_k)} + \|g\|_{C^{1+a}(\Gamma_k)}).$$

Для доведення теореми 1 і теореми 2 використано принцип максимуму для параболічних крайових задач [1], метод “штрафу” [3] і апіорні оцінки [2].

Список літератури

1. Ладыженская О.А., Солонников В.А., Уральцева Н.Н. Линейные и квазилинейные уравнения параболического типа. — М.: Наука, 1967. — 736 с.
2. Фридман А. Уравнения з частними производными параболического типа. — М.: Мир, 1968. — 427 с.
3. Лионс Ж.-Л. Оптимальное управление системами, описываемыми уравнениями с частными производными — М.: Мир, 1972. — 416 с.

Створення навчальних відео-роликів для вивчення предмету “Бази даних”

У сучасному світі, повному різноманітної інформації, надзвичайно важливою є можливість організовувати дані для зручного їхнього використання. В таких випадках у нагоді можуть стати бази даних, у яких описується характеристика даних та взаємозв'язки між даними.

Безсумнівною є необхідність вивчення кожним студентом, котрий прагне стати професіоналом у сфері комп'ютерних технологій, систем управління базами даних. Проте іноді важко знайти якісні відео-матеріали з демонстраціями прикладів та доступними теоретичними поясненнями для повного розуміння принципів роботи з СУБД.

Метою нашої роботи стало створення навчальних відео-роликів із прикладами виконання лабораторних робіт із курсів вивчення основ роботи з базами даних у СУБД MS Access, адже у ній є весь необхідний для початківців спектр функцій. Для запису роликів ми зупинилися на програмі **oCam**, для монтування – **OpenShot**.

Перевагою oCam над іншими подібними програмами є те, що вона може підтримувати усі найпопулярніші формати відео: WMV, MP4, MPEG, FLV, AVI, MOV, TS, VOB; даний пакет є системо-агностичним, тобто здатний працювати як у Windows, так і у Mac, а антивіруси, зокрема Avira, не сприймає запис відео за вірус. При записі відео зі звуком використовуються вбудовані аудіо кодеки (MP3, AAC, FLAC) та запис звуку в режимі реального часу. Корисною властивістю є те, що для розробників залишається доступним bug reporting та забезпечується підтримка подвійного монітора.

Серед основних характеристик обраної нами програми для монтажу OpenShot є підтримка різноманітних форматів відео, аудіо, графіки та кодування відео, що засновані на комплексі

програм та бібліотек для виконання запису, конвертації та пакування у різні формати цифрових аудіо- та відео-матеріалами – Ffmpeg; при монтуванні можна обрізати відео, робити знімки з екрану, змінювати їхні розміри, додавати користувацькі водяні знаки, титри та субтитри, використовувати вбудовані ефекти для переходу між кадрами, пришвидшувати їх чи уповільнювати.

В створених нами роликах присутні детальні пояснення про створення реляційної бази даних з використанням SQL-запитів. У першому ролику є інструкції до створення таблиць, вибору типів даних, ключових полів, створення схеми даних та зв'язків між таблицями, а також пошуку, заміни, сортування, фільтрування полів та переходу по даних. В другому ролику – інформація стосовно створення запитів (простих, на вибірку, з параметрами, перехресного, на створення таблиці, на додавання записів, їх оновлення та видалення). У третьому відео продемонстроване створення форм (автоматичних, за допомогою конструктора, за допомогою майстра). Четвертий кліп містить опис створення звітів(за допомогою майстра звітів, конструктора, та автоматичних звітів). У п'ятому ролику – автоматизація часто виконуваних дій у базі даних за допомогою простих чи складних макросів. Наостанок, у шостому відео – деталі поетапного створення інтерфейсу бази даних.

Отже, актуальність даної роботи на даному етапі полягає у тому, щоб забезпечити початківців у роботі з базами даних детальними поясненнями у вигляді відео-роликів до основ функціонування систем управління базами даних та надати покрокову інструкцію готової бази.

Метод інверсії в задачах

У геометрії велику роль відіграють різні перетворення фігур. У школі детально вивчаються рух і гомотетія, а також їх застосування. Важливою особливістю цих перетворень є збереження ними природи найпростіших геометричних образів: прямі перетворюються в прямі, а кола – в кола. Інверсія являє собою більш складне перетворення геометричних фігур, при якому прямі уже можуть переходити в кола і навпаки. Такий підхід дозволяє застосовувати до задач елементарної геометрії одноманітну методику вивчення. Це, насамперед, відноситься до задач на побудову і до теорії пучків кіл. Зауважимо, що розгляд зазначених розділів елементарної геометрії без застосування інверсії пов'язано із залученням різноманітних, здебільшого досить штучних побудов, що носять частковий характер.

Застосування методу інверсії при розв'язанні задач на побудову й доведення дозволяє вирішити багато завдань, які важко вирішити за допомогою інших методів розв'язання подібних завдань. Суть методу інверсії полягає в наступному. Поряд з даними і шуканими фігурами розглядаємо фігури, інверсні їм або їх частинам. Іноді цього виявляється вже достатньо для знаходження таких зв'язків між шуканими і даними, які потрібні для вирішення завдання. У більшості випадків розв'язання задачі зводиться до побудови фігури, інверсної до шуканої, у припущенні, що вже побудована фігура, інверсна даній [1]. Ця остання задача, при вдалому виборі кола інверсії, може виявитися значно простішою за задану задачу. Побудувавши фігуру, інверсну шуканій, будують шукану фігуру.

Недоліком цього методу є його громіздкість, пов'язана з необхідністю виконати велике число побудов.

Задача. Дано коло Γ і два ортогональних до нього кола g_1 і g_2 без спільних точок (мал. 1). Побудувати коло, ортогональне до всіх трьох кіл Γ , g_1 і g_2 .

Розв'язання. За центр інверсії обираємо точку Q перетину кіл Γ та g_1 , а радіус інверсії може бути довільний. Застосовуючи метод інверсії, маємо, що кола Γ та g_1 перетворюються у ортогональні прямі, а коло g_2 – у коло, ортогональне прямій, що є

образом кола Γ , тобто в коло з центром на цій прямій. Задача звелася до побудови кола (образ шуканого), ортогонального до всіх побудованих образів.

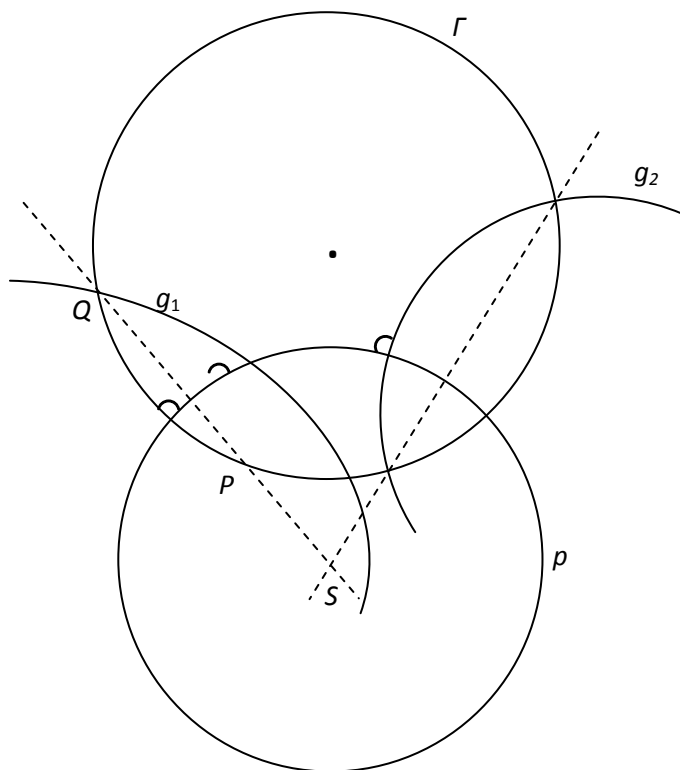


Рис. 1

Список літератури

3. Коба В. І., Нікулін М. А. Найпростіші геометричні перетворення. – К.: Радянська школа, 1978. – 95 с.

ЗМІСТ

<i>Антош Д.</i> Створення блокчейн додатку на базі платформи Ethereum.....	3
<i>Василик А., Рожко О.</i> Розробка розподіленої системи зберігання та перевірки лабораторних робіт студентів.....	5
<i>Возний К.</i> Створення децентралізованого додатку на базі платформи Ethereum.....	7
<i>Волковинський Б.</i> Створення сайту для університетського центру кар'єри ЧНУ ім. Юрія Федьковича	9
<i>Гавалешко В.</i> Атоматизація документообігу навчального процесу	11
<i>Гаврилюк М.</i> Розробка мобільного дододатку для вивчення англійської мови	13
<i>Горбаль Г.</i> Про поліноміальні і нарізно поліноміальні функції	15
<i>Гордій О.</i> Структура фундаментального розв'язку задачі Коші для одного параболічного рівняння з операторами Бесселя різних порядків.....	17
<i>Джигринюк Н.</i> Інформаційна система “Допомога абітурієнту”.....	19
<i>Ільчук Т.</i> Створення програмного забезпечення для плоттера.....	21
<i>Калинюк О.</i> Розробка сайту студмістечка Чернівецького Національного університету ім. Ю. Федьковича.....	23
<i>Клепіковська Ю.</i> Створення системи визначення динаміки стану річкових вод.....	25
<i>Козмєй О., Сидоряк А.</i> Інформаційна система “Військкомат”	27
<i>Крайник О.</i> Властивості фундаментального розв'язку задачі Коші для одного параболічного рівняння з необмеженими коефіцієнтами і операторами Бесселя.....	29
<i>Лукань М.</i> Слабкі R-простори та їх застосування	31
<i>Луник Т.</i> Використання цифрового ідентифікатора об'єкта в наукових публікаціях	33
<i>Мардар А.</i> Синхронізація каналів комп'ютерної мережі при передачі трафіку реального часу.....	35
<i>Мардар О.</i> Аналіз стохастичної моделі волатильності.....	37
<i>Мельник О.</i> Створення навчально - розважальної гри для дітей в середовищі Unity3D	39

<i>Мицкан М.</i> Про рівносильність деяких згорточних спів- відношень	41
<i>Пастула М.</i> Розв’язування задачі хижак-жертва з нело- кальними умовами	43
<i>Пеленяк В.</i> Програмна реалізація емулятора машини з необмеженими регістрами засобами мови Java	45
<i>Полюк Ю.</i> Формування в учнів навичок доведень матема- тичних тверджень при вивченні геометрії	47
<i>Проц С.</i> Децентралізований додаток на основі блокчейн	49
<i>Сушицька В.</i> Розробка ілюстративних матеріалів для вивчення розділу програмування “Бази даних”	51
<i>Чернецький І.</i> Побудова iOS-додатку для побудови графіків і комунікації з плоттером	53
<i>Чігірьов М.</i> Дослідження неперервних константиноїдів	55
<i>Швець Л.</i> Мотивація навчання на уроках математики	57
<i>Штирбу Н.</i> Багатоточкова крайова задача для параболічного рівняння з нерівностями	59
<i>Шуліка А.</i> Створення навчальних відео-роликів для вивчення предмету “Бази даних”	61
<i>Яремак Т.</i> Метод інверсії в задачах	63