

Михайловський Денис Анатолійович
аспірант 2 року навчання
спеціальності 011 Освітні, педагогічні науки
(Теорія і методика вищої освіти)
факультету технологічної та професійної освіти
Глухівського НПУ ім. О. Довженка

МЕТОДИЧНІ ПІДХОДИ ДО НАВЧАННЯ ПРОГРАМУВАННЯ МОВОЮ PROCESSING В НАВЧАЛЬНИХ ЗАКЛАДАХ

Останнім часом у зв'язку з динамічним розвитком ринку праці у сфері ІТ спостерігається зростання потреби у кваліфікованих фахівцях, які володіють сучасними мовами програмування. Відповідно і вивчення програмування в навчальних закладах також набуло все більшої актуальності, де ці знання можуть стати базою для формування світогляду та подальшого профорієнтування учнів у сфері інформаційних технологій [5]. Один із способів впровадження програмування в навчальний процес - це використання мови програмування Processing.

Сам проєкт Processing був ініційований Бенджаміном Фраєм і Кейсі Різом в Media Lab Массачусетського технологічного інституту і до цього часу активно розвивається некомерційною ініціативною групою (Processing Foundation) [1]. Програма Processing розроблена за принципом FLOSS (Free Libre Open Source Software) і розміщенна на сайті <https://processing.org>. Тут кожен може знайти та поділитися своїми цікавими проєктами і знаннями, що присвячені особливостям роботи в Processing [6].

Processing, як програмний засіб для креативного програмування, є частиною цього нового світу генеративного дизайну в ІТ. Генеративний дизайн – розробка дизайну методами програмування, яка дає можливість дизайнерам визначати форми та об'єкти, створювати нові елементи дизайну під час програмування, використовуючи математичні моделі, такі як фрактали, L-системи і множина Мандельброта. Генеративний дизайн використовується в різних галузях дизайну, включаючи мистецтво, архітектуру, комунікації та промисловий дизайн.

Базуючись на Java, Processing має спрощений синтаксис, що робить її дуже доступною для новачків. Головна відмінність від інших мов – її фокус на візуальне програмування.

Processing має особливості, характерні для багатьох мов програмування і тому може слугувати гарним вступом в програмування на інших мовах з використанням інших засобів розробки. Як і будь-яке програмне забезпечення, Processing складається з великої кількості компонентів, що працюють разом. Існує більше 100 бібліотек розширення, за допомогою яких можна застосовувати Processing для опрацювання звуку, досліджень машинного «зору», технічних розрахунків.

Для мови креативного програмування Processing характерні такі особливості: простий і зрозумілий синтаксис мови; можливість працювати в режимі реального часу (live coding), коли можна бачити результат програми одразу після введення рядка з програмним кодом у редакторі; простота установки програми на платформах різних операційних систем; повна «комплектація» мови, включаючи управління сторонніми бібліотеками і сумісність версії для розв'язування графічних, художніх завдань [3].

Враховуючи легкість, зрозумілість і можливості креативного програмування мовою Processing, доцільним є її вивчення учнями в школі. Адже креативне програмування це не просто вивчення коду, але й поєднання технічних знань з творчістю, і як результат – розробка вдалого програмного продукту з цікавим інтерфейсом.

За допомогою Processing є можливість вчитися програмуванню через написання інтерактивних програм, створюючи графічні програми, ігри та візуалізації з мінімальними зусиллями і навчатися основам кодування в ігровій формі. Такий шлях освоєння програмування є цікавим і мотивуючим для учнів старшої школи, оскільки програмування в Processing дає миттєвий результат, всі абстракції стають видимими, алгоритмічні конструкції розглядаються як практичні елементи втілення креативних ідей та творчих експериментів [3].

Для збільшення ефективності навчання програмування в школах з використанням мови Processing пропонуються наступні методичні підходи [4]:

1. **Візуальний підхід:** оскільки Processing орієнтований на візуал, він ідеально підходить для викладання основ програмування через візуальний контекст. Учні можуть легко бачити результати своєї роботи у вигляді графічних зображень та анімацій.
2. **Прогресивне навчання:** починаючи з простих завдань, таких як малювання простих фігур, учні можуть рухатися до більш складних тем, таких як алгоритми та інтерактивність.
3. **Проектна робота:** учні працюють над реальними проектами, такими як створення гри чи інтерактивної анімації. Це підход сприяє розвитку творчості, аналітичного мислення і навичок командної роботи.
4. **Вивчення базових концепцій:** крім практичної роботи, важливо зосередитися на теоретичних аспектах, як-от основи програмування – змінні, умови, цикли та функції.
5. **Ігровий підхід:** ігри можуть бути використані як відмінний інструмент для вивчення програмування, дозволяючи студентам розібратися в коді в ігровій формі.
6. **Інтерактивні онлайн-ресурси:** існує безліч відео-уроків, туторіалів та форумів, присвячених Processing, які можуть служити додатковими ресурсами для навчання.
7. **Групова робота:** робота в командах дозволяє студентам обмінюватися знаннями та досвідом, а також розвивати навички комунікації та взаємодії.
8. **Оцінка та зворотний зв'язок:** регулярний аналіз та оцінка робіт студентів допоможуть їм визначити свої слабкі місця та покращити навички.

Таким чином, використання Processing у навчальних закладах, особливо в контексті креативного програмування і генеративного дизайну, відкриває нові можливості для викладачів та студентів. Методичні підходи, зазначені вище, сприяють не тільки вивченню програмування, але й розвитку критичного мислення, креативності та інших ключових навичок. В цілому, інтеграція таких технологій у

навчальний процес може стати важливим кроком на шляху до формування нового покоління інженерів, дизайнерів та митців.

Список використаної літератури

1. Reas С., Fry В. Processing: A Programming Handbook for Visual Designers and Artists. [Електронний ресурс] – Режим доступу: http://www.hrenatoh.net/curso/processing/processing_benfry.pdf
2. Базурін В. Середовища програмування як засіб навчання учнів основ програмування [Електронний ресурс] – Режим доступу: <https://journal.iitta.gov.ua/index.php/itlt/article/view/1601/1187>
3. Бугаєць Н. Processing як середовище для навчання креативного програмування [Електронний ресурс] – Режим доступу: <http://dspace.tnpu.edu.ua/bitstream/123456789/14519/1/Buhaiets.pdf>
4. Васильєва А. Методичні підходи до навчання програмування за допомогою мови Processing в навчальних закладах програмування [Електронний ресурс] – Режим доступу: <http://eprints.zu.edu.ua/38056/1/%D0%97%D0%B1%D1%96%D1%80%D0%BD%D0%B8%D0%BA%20%D0%BA%D0%BE%D0%BD%D1%84%D0%B5%D1%80%D0%B5%D0%BD%D1%86%D1%96%D1%97%20%D0%9A%D0%BE%D0%BC%D0%BF%27%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D1%96%20%D1%96%D0%B3%D1%80%D0%B8%20%D1%82%D0%B0%20%D0%BC%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BC%D0%B5%D0%B4%D1%96%D0%B0%202023-33-34.pdf>
5. Громко Г. Processing. Основи програмування [Електронний ресурс] – Режим доступу: <https://informatik.pp.ua/pidruchniki/programuvannya/processing-gromko/>
6. Сайт Processing Foundation [Електронний ресурс] – Режим доступу: <https://processing.org/>