

Таким чином, використання комп'ютерних засобів навчання допомагає розвивати у дітей такі вольові якості, як самостійність, зібраність, посидючість.

Перевагами використання комп'ютерних технологій для логопеда є :

- 1) складання звітів і розкладів усіх видів занять в електронному вигляді, що значно зменшує обсяг роботи з паперовими носіями інформації;
- 2) розробка наочно-дидактичного супроводу до занять;
- 3) застосування комп'ютера на логопедичних заняттях як засобу активізації корекційної роботи.

В організації корекційно-розвивальної роботи з дитиною-логопатом можна виділити наступні переваги інформаційних комп'ютерних технологій, використання яких дозволяє:

- 1) підвищити мотивацію дитини до логопедичних занять;
- 2) формувати у дитини активну позицію суб'єкта навчання;
- 3) оволодіти дитині елементарними діями з комп'ютером.

Але треба пам'ятати, що надмірне використання може нашкодити дитині. Постійне використання комп'ютера призводить до підвищення ризику розвитку різних захворювань органів зору, м'язів, суглобів, внутрішніх органів і систем організму, також навколо будь-якої техніки створюється електромагнітне поле, яке тим чи іншим чином позначається на стані здоров'я дитини, тому використання ІКТ повинно буди не основною формою навчання, а допоміжною, тоді навчальний процес буде більш цікавий та безпечний для дитини.

Таким чином, використання інформаційно-комунікаційних технологій дозволяє підійти до питання організації виховного процесу з якісно нового боку, спрощує процес спілкування з учнями та їх батьками. У процесі занять із застосуванням комп'ютера діти вчаться долати труднощі, контролювати свою діяльність, оцінювати результати.

Список використаних джерел:

1. Биков В. Ю. Класифікація засобів навчання : зб. наук. праць. Київ, 2005. 272 с.
2. Буйницька О. П. Інформаційні технології та технічні засоби навчання : навч. посіб. Київ, 2012. 240 с.

## **ПРОБЛЕМА ВИБОРУ МОВИ ПРОГРАМУВАННЯ В ЯКОСТІ ЗАСОБУ ВИВЧЕННЯ ООП**

**Кривонос Олександр Миколайович**

доцент, кандидат педагогічних наук,

доцент кафедри комп'ютерних наук

та інформаційних технологій

krypton@zu.edu.ua

**Галіцький Андрій Вікторович**

здобувач вищої освіти

2 курсу освітньої програми

«Інформатика в закладах освіти»

**Кривонос Михайло Олександрович**

здобувач вищої освіти

1 курсу освітньої програми

«Інформатика в закладах освіти»

Житомирський державний університет

імені Івана Франка

Основи об'єктно-орієнтованого програмування (ООП) є важливою складовою навчання програмуванню і розробці програм. Ось кілька причин, чому важливо опанувати ООП:

**Модульність та структурованість:** ООП допомагає розділити великі програми на менші, більш керовані частини, які називаються класами. Це сприяє підтримці і покращенню коду.

**Поновлюваність:** ООП дозволяє створювати класи, які можна повторно використовувати в різних частинах програми або навіть в інших програмах. Це зменшує дублювання коду та полегшує розвиток.

**Розширюваність:** Ви можете легко додавати нові функції та можливості до існуючих класів без значних змін в іншому коді.

**Поліморфізм:** ООП дозволяє створювати класи, які мають різну реалізацію для спільних методів. Це сприяє загальному коду та більш зручному використанню.

**Спадкування:** Можливість створення нових класів на основі існуючих допомагає зберегти та розширити функціональність.

**Абстракція:** ООП сприяє створенню абстрактних класів та інтерфейсів, що дозволяє розкрити лише необхідну інформацію для користувача класу.

**Покращена організація:** ООП допомагає логічно структурувати код і краще розуміти, як різні частини програми взаємодіють між собою.

**Зменшення ризику помилок:** Код, розроблений з використанням ООП, часто менше схильний до помилок, так як він структурований та більш понятний.

**Підтримка командної роботи:** ООП сприяє розділенню роботи між різними членами команди, оскільки різні класи можуть бути розроблені та підтримані окремими програмістами.

**Популярність:** Багато сучасних мов програмування (таких як Java, C++, C#, Python) побудовані на основі ООП, тому знання цієї парадигми допомагає розуміти і працювати з різними мовами.

Загалом, знання основ ООП сприяє розробці більш структурованих, підтримуваних і розширюваних програм, що є важливими навичками для програміста.

Багато сучасних мов програмування підтримують парадигму об'єктно-орієнтованого програмування (ООП) або принаймні мають можливості для роботи з об'єктами. Ось деякі з найпопулярніших мов програмування, які підтримують ООП:

- Java є однією з найпопулярніших мов програмування для ООП. Усі програми Java базуються на об'єктах, і мова надає великий набір бібліотек для роботи з об'єктами та спадкуванням.

- C++ - це мова, яка поєднує можливості низькорівневої мови C з ООП. Вона підтримує створення класів, об'єктів, спадкування, інкапсуляцію та багато інших концепцій ООП.

- Python підтримує ООП і є високорівневою мовою програмування. Вона дозволяє створювати класи, об'єкти і використовувати спадкування та поліморфізм.

- C# - мова програмування від Microsoft, яка в основному використовується для розробки додатків для платформи .NET. Вона має потужну підтримку ООП, включаючи класи, інтерфейси, спадкування тощо.

- Ruby - це мова програмування, яка відома своєю простотою та зручністю для розробників. Вона використовує концепції ООП, такі як класи та об'єкти, і підтримує метапрограмування.

- PHP - це мова, яка широко використовується для розробки веб-додатків. Вона підтримує об'єктно-орієнтований стиль програмування і має можливості для роботи з класами та об'єктами.

- Swift - це мова програмування, розроблена Apple для розробки додатків під iOS та macOS. Вона має сучасний підхід до ООП та підтримує класи, структури та протоколи.

Це лише декілька прикладів мов програмування, які підтримують ООП. Багато інших мов також використовують цю парадигму або мають можливості для створення об'єктів та використання класів.

Вибір мови програмування для початку знайомства з об'єктно-орієнтованою парадигмою (ООП) залежить від ваших конкретних мет і інтересів. Ось кілька популярних мов програмування та рекомендації щодо того, з якої мови краще почати це Python, Java, JavaScript, Ruby або C#.

Оберіть мову програмування відповідно до ваших інтересів і мети. Важливо розуміти, що концепції ООП переносяться з однієї мови на іншу, тому знання ООП може бути корисним навіть після переходу до іншої мови.

Існує декілька сайтів та платформ, які можуть допомогти вам вивчати об'єктно-орієнтоване програмування (ООП) в ігровій формі. Ось деякі з них:

- CodeCombat (codecombat.com):** CodeCombat пропонує ігри, в яких ви можете програмувати, щоб керувати персонажем у вигляді героя і розгадувати завдання. Це весело і візуально захоплююче середовище для вивчення ООП та інших мов програмування.

• **Scratch (scratch.mit.edu):** Scratch - це візуальна мова програмування, яка допомагає вам створювати ігри, анімацію та інтерактивні проекти, використовуючи блоки коду. Вона ідеально підходить для початківців.

• **RoboZZle (robuzzle.com):** RoboZZle - це гра, в якій ви програмуєте робота, щоб розгадати головоломки. Ви вчитеся вирішувати завдання за допомогою ООП-концепцій.

• **CheckiO (checkio.org):** CheckiO - це платформа для вивчення Python через розв'язання головоломок та завдань. Ви використовуєте ООП-принципи для вирішення задач на цій платформі.

• **Codingame (codingame.com):** Codingame пропонує ігри, де ви програмуєте, щоб керувати персонажем або розв'язувати завдання. Вона також включає конкурси та завдання для програмістів на будь-якому рівні.

• **HackerRank (hackerrank.com):** HackerRank - це платформа для розв'язування завдань у програмуванні, включаючи ООП. Ви можете змагатися з іншими програмістами та вирішувати головоломки.

• **Codewars (codewars.com):** Codewars пропонує вирішувати завдання, використовуючи ООП-концепції. Ви можете обирати завдання різного рівня складності.

З цими платформами ви можете навчати ООП, вирішуючи завдання та головоломки в ігровій формі. Це допоможе вам отримати практичний досвід і зрозуміти концепції ООП краще.

Розглянемо більш детально одну з платформ. Codingame (codingame.com) - це платформа для навчання програмуванню через гру. Вона пропонує різні ігри та головоломки, в яких ви можете використовувати свої навички програмування для вирішення завдань. Ось докладніша інформація про Codingame:

• **Мови програмування:** Codingame підтримує різні мови програмування, включаючи Python, Java, C++, JavaScript, C#, Ruby та інші. Ви можете вибрати мову, яка вам найбільше підходить або яку ви хочете вивчити.

• **Завдання та головоломки:** Платформа пропонує різні завдання, включаючи головоломки, гри та конкурси. Завдання розрізняються за рівнем складності, від початкового до експертного, що дозволяє користувачам вибирати завдання, які відповідають їхнім навичкам.

• **Спільнота програмістів:** Codingame має активну спільноту програмістів, які змагаються між собою та спільно розв'язують завдання. Ви можете брати участь у різних конкурсах та викликах, а також спілкуватися з іншими користувачами.

• **Навчання ООП:** Платформа пропонує завдання та гри, які допомагають вивчити і відточити навички об'єктно-орієнтованого програмування (ООП). Ви можете розробляти програми, використовуючи ООП-підходи, і практикувати їх на Codingame.

• **Створення власних завдань:** Користувачі також мають можливість створювати власні завдання і ділитися ними з іншими. Це дозволяє розвивати власні навички та вчити інших.

•Безкоштовний доступ: Базовий рівень доступу до Codingame є безкоштовним, і ви можете користуватися багатьма функціями та завданнями. Є також платний план, який надає додаткові можливості, такі як розширений аналіз та підтримка.

Загальна ідея Codingame полягає в тому, щоб навчати програмуванню через захоплюючі ігри та завдання. Вона дозволяє вам навчатися, вирішуючи практичні завдання та конкурси, і вдосконалювати свої навички програмування.

Список використаних джерел:

1. Базурін В. М. Середовища програмування як засіб навчання учнів основ програмування / В. М. Базурін // Інформаційні технології і засоби навчання. - 2017. - Т. 59, вип. 3. - С. 13-27. [Електронний ресурс]. – Режим доступу: [http://nbuv.gov.ua/UJRN/ITZN\\_2017\\_59\\_3\\_4](http://nbuv.gov.ua/UJRN/ITZN_2017_59_3_4)

2. Климчук В.В. Методика вивчення об'єктно-орієнтованого програмування. Інформаційно-комунікаційні технології в освіті/ В.В. Климчук – К.: УОВЦ «Оріон», 2018. [Електронний ресурс]. – Режим доступу: <https://e-journals.npu.edu.ua/index.php/ikt/article/view/142> 2018, 5.

3. Кравець П. Об'єктно-орієнтоване програмування: навч. посібник / П.О. Кравець. – Львів: Видавництво Львівської політехніки, 2021. – 624 с

4. Шевчук, П.Г. Проблема вибору мови та середовища програмування в якості засобу навчання. / П.Г. Шевчук, Звітна наукова конференція Інституту інформаційних технологій і засобів навчання АПН України: Матеріали наукової конференції, 2010,- С.30-31.

## ПЕРСОНАЛІЗАЦІЯ ЯК ПІДХІД ДО НАВЧАННЯ МАЙБУТНІХ ІТ-ФАХІВЦІВ

**Сікора Ярослава Богданівна,**  
кандидат педагогічних наук, доцент  
Житомирський державний університет  
імені Івана Франка, Україна  
[iaroslava.sikora@gmail.com](mailto:iaroslava.sikora@gmail.com)

Розвиток інформаційно-комунікаційних технологій та підвищення доступності технічних засобів навчання в освіті, а також паралельне зростання кількості та доступності інформації для здобувачів освіти призвели до необхідності пошуку нових рішень, що дозволяють ефективно використовувати перелічені досягнення в освітньому процесі. Для відповідності професійної підготовки поточним вимогам до майбутнього ІТ-фахівця необхідно формувати компетентності, що дозволяють здобувачам освіти у майбутньому самостійно організовувати процес саморозвитку з використанням усіх наявних ресурсів та видів освіти (формальна, неформальна чи інформальна). У зв'язку з цим процес індивідуалізації в освіті, орієнтований на здобувача зі збільшенням його