

Дослідження 12. Як знайти спільну мову з виконавцем?

Методичний коментар до уроку 16 “Виявлення та виправлення помилок у програмах”

На уроці: передбачення результату виконання алгоритму. Як заздалегідь перевірити алгоритм, щоб під час його виконання було менше помилок? Робота в парах: перевірка алгоритмів.

У наступних дослідженнях уже не вивчатимуться нові команди виконавця “Кенгуру”. Зосереджуємося на практичній роботі — ретельному продумуванні алгоритмів та якісному написанні програм, пошуку і виправленні помилок у них; поглибленні вивченого матеріалу, напрацюванні досвіду створення алгоритмів і програм, досягненні майстерності в програмуванні.

Тож на уроках варто ставити акцент на спокійній, ретельній, виваженій роботі над програмами в середовищі виконавця “Кенгуру” відповідно до рівня знань і вмінь, бажань дітей, за потреби — на індивідуальній роботі з дітьми (як з тими, у кого не все виходить, так і з тими, у кого програмування — найулюбленіша тема інформатики). Відбувається повторення, поглиблення, узагальнення та систематизація матеріалу теми, відпрацювання вмінь і навичок, набуття досвіду роботи в середовищі програмування. Дайте кожній дитині можливість відчувати своє зростання, упевненість у власних силах.

Опитування “[Помилки в програмах](#)” (до уроку) передусім визначає ставлення дітей до помилок, розуміння важливості їх виправлення, готовність ретельно працювати над їх усуненням, прагнення зробити свою програму якомога кращою. Також це опитування показує, наскільки учням буде цікава тема цього уроку.

Опитування також містить інформацію для роздумів дітей, адже не всі задумувалися, що програмування — це ретельна і зважена робота над різними ситуаціями, у яких має працювати програма. Іноді таку роботу не видно, але її треба обов’язково завершити, протестувати програму і впевнитися, що за будь-яких обставин програма спрацює правильно і не виникне непередбачуваної ситуації.

У запитанні “Який для тебе найкращий спосіб знайти помилку?” містяться різні варіанти усунення помилок. Не всі вони рівнозначні та можуть дати результат, на який ми сподіваємося. Тож вибір одного з варіантів може вказати на характер дитини, прагнення до самостійності, серйозність ставлення до роботи тощо.

Неслухняний виконавець

Запропонуйте дітям прочитати невеличку історію на с. 43 підручника й обговорити запитання, запропоновані в завданнях.

Завдання 1 (підручник с. 43). Чи доводилося вам сердитися на комп’ютерного виконавця, іншу програму чи пристрій, який “не слухається”? Чи вдалося вам порозумітися з ним? Як саме це відбулося?

Завдання 2 (підручник с. 43). Як ви вважаєте, чи справді виконавець “нерозумний”? Поясніть свою думку. Чи міг виконавець не слухатися програми, яку склав Олексійко?

Діти діляться одне з одним власним досвідом, радяться, шукають найкращий варіант приборкування програми чи пристрою.

У роздруківці “[Якими бувають помилки](#)” дітям потрібно знайти помилки в добре знайомій їм програмі малювання квадрата зі стороною один крок. Тому діти можуть легко уявити дії виконавця “Кенгуру”, знайти помилки та виправити їх без взаємодії з комп’ютером. У роздруківці є правильні програми і такі, де виконавець правильно малює квадрат, але в них є помилки. Справді, зараз ця помилка не принципова, але коли малювання квадрата стає частиною іншої програми, то такі помилки викликають інші недоліки в роботі програми, адже виконавець завершує малювання не там, де ми планували, і стоїть не так. А якщо таких нестандартних рішень вимагає певна ситуація, то маємо свідомо писати програму і розуміти, навіщо ми ставимо ту чи ту команду саме так. Тож зверніть увагу дітей на те, що помилка може виникати навіть тоді, коли результат правильний.

Обговоріть з дітьми, на що варто звертати увагу, коли помилка сталася, що допоможе швидше зрозуміти, де саме помилка, чому програма працює неправильно і як виправити недоліки.

У роздруківці також пропонується дітям рейтингувати помилки, тобто визначити, які з них найсуттєвіші. Але як же зрозуміти значущість помилки? Дітям здається, що чим результат менше схожий на квадрат, тим значуща помилка. Зауважте, що будь-яка помилка вимагає виправлення, адже ви ніколи не знаєте, до чого може призвести навіть найменший недолік у певній ситуації. Тож насправді в програмуванні немає маленьких і великих помилок. Кожна помилка вимагає опрацювання і виправлення. Просто є помилки, яких до певного часу не видно, але в певній ситуації вони можуть “вибухнути”.

Роздруківка “[Знайди помилку в програмі](#)” продовжує думку попередньої і дає змогу обговорити з дітьми, чи справді в програмі немає помилки, якщо Кенгуру може її виконати. Знайти помилки в програмах у цій роздруківці дітям може бути складніше, тому запропонуйте їм винайти зручні способи пошуку та виправлення помилок, перевірки правильності роботи програми без взаємодії з комп’ютером. Така робота сприяє більшому розумінню сутності команд, адже за комп’ютером дитина може, не думаючи над змістом програми, виправляти помилки, нашаровуючи команди одну за одною. А без допомоги гаджета дитина може діяти вдумливіше, стриманіше і знайде причину помилки та виправить її.

Можна сформувати кілька груп дітей і дати їм попрацювати з різними роздруківками. Потім попросити кожну групу презентувати роботу, розповісти, як шукали помилки, та зробити висновки. Але краще, щоб діти опрацювали обидва завдання в роздруківках, потренувалися знаходити помилки та зробили власні висновки.

Розкажіть дітям, що робота над помилками — невід’ємна частина праці програміста. Адже помилки в програмах можуть призвести до серйозних наслідків у роботі різних організацій — банків, заводів, електростанцій, лікарень тощо. Причому проблема може виникнути там, де її не чекали.

Наприкінці ХХ століття виникла “Проблема 2000 року” — можливості неправильної роботи програмного забезпечення через перехід від 1999-го до 2000 року. Цю проблему вирішували не лише програмісти різних країн, а й уряди багатьох країн світу.

Проблема пов’язана з тим, що розробники програмного забезпечення, випущеного в ХХ столітті, іноді використовували два знаки для представлення року в датах, перші два вважалися рівними 19. Тому рік 2000 міг інтерпретуватися старими програмами як 1900,

а це могло призвести до серйозних збоїв у роботі систем управління технологічними процесами, фінансових програм тощо. Тож необхідно було перевірити наявне програмне забезпечення, і тому створювалися відповідні комісії. Наприклад, МОЗ України у квітні 1999 року видало наказ про створення галузевої комісії.

Нещодавно помилки в оновленнях популярних сервісів в ОС “Андроїд” у смартфонах призвели до збоїв у пошуку в Google, роботі електронної пошти, хмарних сервісів, якими користуються люди в усьому світі. Тож великі компанії втратили гроші, власники смартфонів стикнулися з тим, що не можуть вчасно виконати певну роботу. Уявіть, як почувалися ці люди. А якщо збої виникають на космічних кораблях, МКС, атомних станціях тощо? Тож програма має бути складеною так, щоб правильно працювати в будь-яких ситуаціях, а кожний програміст має уважно ставитися до усунення найменших помилок.

Запропонуйте дітям потренуватися знаходити і виправляти помилки також в алгоритмах у завданнях 1–2 (с. 28) [зошита з інформатики](#) (автори — О. Андрусич, І. Стеценко).

Перетворюємося на Кенгуру

Час підсумувати розмову про виявлення та усунення помилок в алгоритмах і програмах. Запропонуйте дітям прочитати текст на с. 44 підручника та виконати завдання 3.

Завдання 3 (підручник с. 44). Виконайте на аркуші в клітинку програму для Кенгуру.

початок	Зроби крок
Зроби крок	Зроби крок
Повернись ліворуч	Повернись ліворуч
Повернись ліворуч	Зроби крок
Стрибни	Стрибни
Повернись ліворуч	кінець

- Який вийшов результат?
- Чи в усіх вийшов однаковий результат?
- Як варто перевірити правильність виконання завдання, щоб не було помилки? Чому саме так?
- Як діяли б ви, якби вам потрібно було написати таку програму? Чому ви діяли б саме так?

Запропонуйте дітям скласти власну програму та порівняти її з програмою в підручнику.

Завдання 4 (підручник с. 44). Складіть у зошиті програму для Кенгуру, виконавши яку, він “напише” слово “СОН”. Обміняйтеся зошитами з однокласником чи однокласницею та перевірте програми одне одного, виконавши їх на аркушах у клітинку.

Роздруківка-пам’ятка [“Як знайти помилку в програмі”](#) допоможе вам підбити підсумок.

Запропонуйте дітям виконати алгоритм у завданні 4 (с. 29) [зошита з інформатики](#) (автори — О. Андрусич, І. Стеценко), написати алгоритми створення послідовностей

геометричних фігур (завдання 5, 6 на с. 29 зошита) і попрацювати в парах для перевірки правильності роботи.

Робота з комп'ютером. Виправляємо помилки

Діти працюють у середовищі програмування виконавця “Кенгуру”: створюють програми із циклами, тестують їх і виправляють помилки.

Для практичної роботи можна скористатися завданнями з підручника чи картками чотирьох рівнів (роздруковка “[Картки: виправляємо помилки](#)”). Частина завдань можна дати дітям додому, а в класі на наступному уроці вони за комп'ютером самостійно можуть перевірити, наскільки правильно виконали завдання.

За бажанням діти можуть виконати додаткове завдання — дописати чи вдосконалити програму, виконавши бонусне завдання. Це не додаткове завдання, а таке, де потрібно змінити вже написану програму — замінити лінійну програму на програму із циклом, щось домалювати тощо.

У дітей уже є досвід вибору завдань, тому вони можуть об'єктивно оцінювати власні сили і вибирати завдання собі до снаги. Не стимулюйте дітей обирати прості завдання з додатковими бонусними балами за складність. Ставте їм більше бонусів за виконання додаткових завдань вищих рівнів складності тощо.

Дітям важливо мати об'єктивне уявлення про рівень власних знань і вмінь, правильно оцінювати власні сили, бути впевненими в тому, що можна розв'язати навіть те завдання, яке спершу здається непростим. Пояснюйте дітям, що не варто брати легкі завдання лише тому, що страшно не виконати складніші. Самостійно вибрати завдання для дітей дуже важливо ще й тому, що так зникає страх, що завдання буде складним. Зауважайте, що діти також можуть вибрати інше завдання, якщо не зможуть виконати попереднє.

Спостереження за вибором завдань та їх розв'язуванням, зіставлення вибору учнів та їхніми успіхами — потужний інструмент формування оцінювання. Адже вибір дітей покаже і невпевненість у власних силах, і невміння їх розраховувати, і брак чесної інформації про власні знання, і незнання себе, і ще багато іншого.

За потреби на цьому уроці та наступних можна також пропонувати дітям картки з попередніх уроків: діти можуть або ще один раз виконати те саме завдання, і зробити його з іншою якістю, або вибрати складніше завдання. Тоді можна буде бачити зростання дітей протягом кількох уроків.

Удома надайте дітям змогу самостійно створити опитувальник, який перевірятиме, як учні вміють програмувати в середовищі виконавця “Кенгуру”. Діти можуть самостійно вибрати зручну для себе форму — написати все на окремому аркуші, використати текстовий редактор чи працювати в Google Forms. Акцентуйте не на формі, а на змісті опитувальника. Виконуючи таке завдання, діти наочно бачать, що складати опитувальники так, щоб завдання були зрозумілі іншим та мали рішення, непросто. На уроці діти можуть помінятися опитувальниками та виконати запропоновані завдання, а потім перевірити завдання одне одного та доброзичливо обговорити опитувальники.

Завдання 3 (с. 28) [зошита з інформатики](#) (автори — О. Андрусич, І. Стеценко) на повторення можна дати як домашнє.