

РІВНЕНСЬКИЙ ДЕРЖАВНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ

Факультет математики та інформатики

Кафедра інформаційно-комунікаційних технологій та методики викладання інформатики

«До захисту допущено»

Завідувач кафедри

Войтович І. С.

(підпис)

«__»_____ 202__р. протокол №

ГОЙДА ВЯЧЕСЛАВ ОЛЕГОВИЧ**КВАЛІФІКАЦІЙНА РОБОТА****ОСВІТНЯ РОБОТОТЕХНІКА****ЯК ЗАСІБ РОЗВИТКУ МИСЛЕННЯ УЧНІВ**

014 Середня освіта (за спеціалізаціями)

014.09 Середня освіта (Інформатика)

Подається на здобуття другого (магістерського) рівня вищої освіти

Кваліфікаційна робота містить результати власних досліджень і немає запозичень з праць інших авторів без відповідних посилань

(підпис)Гойда В.О.
(прізвище, ініціали)

Науковий керівник: Павлова Наталія Степанівна, професор, доцент кафедри інформаційно-комунікаційних технологій та методики викладання інформатики, кандидат педагогічних наук

ЗМІСТ

ВСТУП 4

РОЗДІЛ І. ТЕОРЕТИЧНІ ОСНОВИ ВИКОРИСТАННЯ SCRATCH ТА SPIKE PRIME У РОБОТОТЕХНІЦІ	7
1.1. Загальні поняття про робототехніку	7
1.2. Scratch та Spike Prime	12
1.3. Принципи роботи Scratch та Spike Prime	16
1.4. Типи мислення у школярів	19
РОЗДІЛ 2. SCRATCH ДЛЯ РОБОТОТЕХНІКИ	24
2.1 Історія та основні концепції Scratch	24
2.2. Використання Scratch для програмування роботів	26
2.3. Як скласти робота засобами роботехнічного конструктора	29
2.4. Приклади проектів з використанням Scratch	32
2.5. Переваги та недоліки програми Scratch	33
РОЗДІЛ 3: SPIKE PRIME ДЛЯ РОБОТОТЕХНІКИ	37
3.1. Комплект Spike Prime та його застосування	37
3.2. Програмування Spike Prime для створення роботів	38
3.3. Як скласти робота засобами роботехнічного конструктора Spike Prime	39
3.4. Приклади проектів з використанням Spike Prime	41
ВИСНОВКИ	47
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	48
ДОДАТКИ	51

ВСТУП

Робототехніка є однією з найбільш інноваційних галузей сучасної освіти, модернізуючи її технологічний розвиток. За допомогою програмованих роботів, здобувачі освіти різного віку мають можливість вивчати наукові концепції, формувати навички програмування та інженерії, а також розвивати «soft skills». Однак, для досягнення цих цілей необхідно виважено дібрати доцільні інструменти та підходи до навчання робототехніці.

Серед різноманітних платформ для робототехніки, найбільш популярними та доступними і такими, що використовуються у закладах загальної середньої освіти є Scratch та Spike Prime. Scratch – це мова програмування та інтерактивне середовище розвитку учнів, що дозволяє створювати програми, ігри та анімації шляхом з'єднання блоків. Spike Prime, відомий своєю простотою та широкими можливостями для навчання робототехніці шляхом вивчення основ інженерії та програмування.

Ця магістерська робота присвячена дослідженню використання Scratch та Spike Prime у робототехніці, а також порівняльному аналізу цих платформ. Метою дослідження є розгляд можливостей, обмежень та навчального потенціалу кожної з цих платформ, а також визначення їхнього впливу на освітній процес та розвиток у здобувачів освіти відповідних навичок.

Серед завдань дослідження: розглянути історію розвитку Scratch та Spike Prime, описати їх основні концепції, навести приклади проектів, які реалізовані за допомогою цих платформ. Також провести порівняльний аналіз інтерфейсів та інструментів, оцінити ефективність, у тому числі навчальний потенціал кожної платформи, визначити переваги та обмеження їх використання у навчанні робототехніці.

Тема роботи – «Освітня робототехніка як засіб розвитку мислення учнів».

Мета роботи – теоретично розглянути робототехніку як освітню дисципліну, практично підтвердити розвиток мислення учнів шляхом складання робота і його програмування.

Об'єкт роботи – освітня робототехніка як інноваційна галузь, що поєднує інженерію, програмування та психологію.

Предмет роботи – Scratch та Spike Prime як інструменти вивчення робототехніки та розвитку на цій основі мислення учнів.

Відповідно до мети, об'єкта і предмета було визначено такі **завдання роботи**:

1. Спираючись на аналіз вітчизняної та зарубіжної літератури з обраної проблематики, визначити теоретичні засади освітньої робототехніки.
2. На підставі аналізу наукових джерел та практичного досвіду дослідити особливості складання і програмування роботів учнями закладів загальної середньої освіти.
3. Навести приклади складання і програмування роботів засобами SCRATSH, SPIKE PRIME, спираючись на особливості вікового та індивідуального розвитку учнів.

Апробація результатів. Основні положення та результати дослідження доповідались та обговорювались на:

- XIV Всеукраїнській науково-практичній конференції «Інформаційні технології в професійній діяльності» (18.11.2020 р., м. Рівне);
- II Всеукраїнській науковій конференції «Підготовка педагогів до професійної діяльності в умовах змішаного навчання» (30.05.2023 р. м. Рівне);
- XVI Всеукраїнській науково-практичній конференції «Інформаційні технології в професійній діяльності» (1.11.2023 р., м. Рівне).

Публікації. Опубліковано тези доповідей:

- «Робототехніка засобами SCRATSH» у збірнику матеріалів XIV Всеукраїнської науково-практичної конференції «Інформаційні технології в професійній діяльності» (1.11.2021 р., м. Рівне, с. 96);
- «Робототехніка, як об'єкт вивчення у закладах освіти» у збірнику матеріалів II Всеукраїнської наукової конференції «Підготовка педагогів до професійної діяльності в умовах змішаного навчання» (30.05.2023 р. м. Рівне, с. 98);

- «Робототехніка в новій українській школі» у збірнику матеріалів XVI Всеукраїнської науково-практичної конференції «Інформаційні технології в професійній діяльності» (1.11.2023 р., м. Рівне, с. 14).

Структура роботи: робота містить вступ, три розділи, висновки, список джерел із 25 праць, додатки, у тому числі, 47 сторінок основного тексту, 20 рисунків, 3 таблиці.

РОЗДІЛ I. ТЕОРЕТИЧНІ ОСНОВИ ВИКОРИСТАННЯ SCRATCH ТА SPIKE PRIME У РОБОТОТЕХНІЦІ

1.1. Загальні поняття про робототехніку

Робототехніка – це галузь науки і техніки, яка представляє створення, розробку, виробництво роботів і їх застосування у різних галузях; область техніки, пов'язана з розробкою і застосуванням роботів і комп'ютерних систем управління ними. Існує багато типів робототехнічних пристроїв, в тому числі роботи-маніпулятори, мобільні роботи, крокуючі роботи.

Робототехніка передбачає проектування, конструювання та програмування механізмів – роботів, що мають модульну структуру і побудовані на мікропроцесорах. Робот – це автоматизована машина, здатна виконувати певні завдання, а також самостійно приймати рішення у процесі виконання цих завдань. Історія робототехніки налічує кілька тисячоліть. У робота є процесор, датчики для сприйняття його середовища, а також двигуни та приводи для переміщення кінцівки або коліс. Він може говорити, робити інші звуки, або спалахувати з вогнями та кольорами у відповідь на навколишнє середовище відповідно до інструкцій. Роботам потрібно вміння слідувати запрограмованим інструкціям, а не просто керувати дистанційно.

Перші згадки про роботи зустрічаються в давньоєгипетських, давньогрецьких і давньоримських рукописах. У Середньовіччі роботи були відомі в Європі, Китаї та Японії. Сучасна робототехніка зародилася в середині ХХ століття. Одним із основоположників робототехніки є американський інженер і винахідник Джозеф Енгельбергер. У 1954 році він представив першого промислового робота, який міг виконувати складні операції процесу зварювання.



Рис. 1.1. Перший промисловий робот Джозефа Енелбергера

У 1960-х роках робототехніка активно розвивалася, але в окремих галузях. 1970-і роки характеризуються широким застосуванням роботів у промисловості. В 1980-х роках роботів почали використовувати у медицині. В 1990-х виникли роботи з самонавчанням. У 2000-х робототехніка стала ключовою у сільському господарстві, будівництві, логістиці та обслуговуванні (Riener, 2016, 49).

Сьогодні роботи використовуються в багатьох галузях. У міру того, як технологія продовжує розвиватися, з'являється ще більше застосувань роботів у нашому житті. Одним із ключових напрямків розвитку робототехніки є автономія – робот приймає рішення самостійно, без втручання людини.

Іншим важливим напрямком розвитку робототехніки є штучний інтелект (ШІ). ШІ надає роботам здатність навчатися та адаптуватися до нових ситуацій, виконувати більш складні завдання, наприклад, розпізнавання образів та голосу. Однак важливо використовувати робототехніку відповідально, щоб уникнути негативних наслідків, таких як втрата робочих місць або ризик безпеки. Варто розглянути важливі наукові досягнення та технологічні інновації, що суттєво змінили парадигму робототехніки в сучасному світі та в освіті, зокрема.

Спершу, слід звернути увагу на прориви у галузі м'яких роботів. Одним із ключових досягнень було створення роботів на основі м'яких матеріалів і актуаторів, що дозволило побудувати біоморфні структури, що можуть

взаємодіяти з навколишнім середовищем більш ефективно (Laschi & Cianchetti, 2016, с. 4).

Друга важлива тенденція полягає в поєднанні роботів та ШІ. Розвиток нейромережових та навчальних алгоритмів дозволяє роботам більш ефективно реагувати на незрозумілі ситуації, виконувати складні завдання (Levine & Finn, Darrell, 2016, с. 1-40). Ця інтеграція допомагає роботам стати більш автономними та здатними вирішувати проблеми в реальному часі.

Нейронні мережі – це тип ШІ, який містить нейрони людського мозку. Вони складаються з вузлів, які називаються нейронами, які з'єднані між собою. Кожен нейрон отримує вхідні дані від інших нейронів і опрацьовує їх, використовуючи функцію активації. Вихідний сигнал нейрона потім передається іншим нейронам.

Нейронні мережі можуть використовуватися для вирішення широкого спектру завдань, таких як розпізнавання образів, розпізнавання голосу, машинний переклад та прогнозування. Вони є особливо ефективними для вирішення завдань, які вимагають навчання на великих наборах даних.

Таким чином, «Робототехніка» – це прикладна наука, що займається розробкою автоматизованих систем.

Освітня робототехніка – це новий міждисциплінарний напрямок навчання, що інтегрує знання з фізики, технології, математики, інформатики та ІКТ, залучаючи в процес інноваційного науково-технічної творчості учнів різного віку. Освітня робототехніка популяризує науково-технічну творчість, підвищує інтерес до інженерних професій, розвиває навички практичного вирішення актуальних інженерно-технічних завдань і роботи з технічними пристроями.

Існує декілька напрямків вивчення робототехніки: *free model* (творче конструювання роботів без поглиблення у механізми та алгоритми); *half models* (вивчення вузлів, датчиків, сенсорів, процесорів та інших елементів в одній конструкції, наприклад, в інтерактивному роботі); *one model* (проектування, складання, програмування робота та його різностороннє вдосконалення) (рис. 1.2.) (Робототехніка у середній школі [Електронний ресурс], 2023).

Рис. 1.2 Ідеї вивчення робототехніки у шкільному курсі інформатики С. С. Баранов розроблені рішення для навчання освітньої робототехніки поділяє на такі основні типи:

- Інтерактивні роботи з функцією програмування:
 - програмовані роботи: крокові, колісно-гусеничні, мультикоптери.
 - модульні роботи.
- Набори компонентів для створення роботів або детального дос-лідження окремих компонентів системи:
 - конструктори для створення й програмування роботів, або робототехнічні платформи;
 - схемотехнічні набори з функцією програмування;
 - конструктори для фізичних експериментів без функції програмування (Баранов, 2020).

На ринку робототехніки є різні платформи, спеціально розроблені для впровадження в освітній процес. Метою введення робототехніки в навчальні предмети є формування в учнів обізнаності щодо моделювання, конструювання, побудова і програмування роботизованих систем, а також встановлення міжпредметних зв'язків інформатики з іншими галузями.

Аналіз стану навчання робототехніки в закладах освіти показав, що її вивчення здійснюється: окремими темами на уроках інформатики або технологій у закладах загальної середньої освіти; за спеціально розробленими програмами гурткових занять у позашкільних закладах за інтересами дітей з урахуванням вікової категорії; окремою спеціальністю чи дисциплінами «Освітня

робототехніка», «Робототехнічні системи», «Stem-освіта», «Штучний інтелект» та ін. у закладах вищої освіти (Гойда & Павлова, 2023, с.98).

Проєктування і створення різних конфігурацій роботів та їх програмування засобами мови програмування підвищують в учнів інтерес до технічної галузі знань, сприяють розвитку soft skills (генерування творчості і уяви, застосування мисленнєвих прийомів, командна робота), поглиблюють знання про алгоритми та їх програмування у певних середовищах.

Однак, як стверджує О. М. Кривонос, в Україні недостатньо приділяється уваги розвитку робототехніки в освітньому процесі. Зокрема, навчання освітньої робототехніки відбувається: у школах в рамках навчання інформатики, ІКТ, технологій, фізики (як модулів або окремих тем); на факультативах та гуртках в закладах середньої загальної освіти у позаурочний час (включаючи підготовку учнів до участі в роботі МАН, змаганнях з робототехніки); у позашкільних навчальних закладах (Кривонос, 2023).

Освітня робототехніка динамічно розвивається, масштабно впроваджується в освіту, тому вчителі повинні спиратися на готовність і здатність поглиблювати й оновлювати власні знання й уміння в цій галузі (Гойда & Павлова, 2023, с. 14). Для того, щоб навчитися створювати роботів і писати для них програми, учням необхідно засвоїти додаткові знання з механіки, радіоелектроніки та програмування. Одних знань для створення роботів недостатньо. Учням необхідно отримати відповідні уміння та навички. У результаті навчання як в області механіки, так і в області електроніки, учні повинні: навчитися визначати функції та технічні характеристики компонентів робота (механічних та електронних); вміти розраховувати і виготовляти необхідні механічні, електронні та електричні компоненти; вміти зобразити зовнішній вигляд певної конструкції і пояснити іншим людям принцип дії, довести правильність її роботи.

1.2. Scratch та Spike Prime

Сучасний світ опрацювання даних і повідомлень вимагає від освіти активного включення в навчання мислення та робототехніки. Одним з інструментів для досягнення цих цілей є програма Scratch. Розглянемо використання Scratch для навчання робототехніки та проаналізуємо вплив цієї мови програмування на розвиток мислення та творчості в учнів.

Робототехніка є галуззю, яка поєднує в собі програмування та фізичний світ. Однак, навчання робототехніці у поєднанні з програмуванням є складним завданням, особливо для учнів.

В області програмування роботів учні повинні: вміти отримувати та обробляти вхідні сигнали робота (від датчиків, кнопок та ін.); програмним способом управляти вихідними пристроями робота (механізмами, пристроями індикації, звуковими пристроями та ін.); створювати програми відповідно до технічного завдання (Кузьменко, 2023).

Програма Scratch створена для полегшення процесу навчання, роблячи його доступним і зрозумілим для широкої учнівської аудиторії. Візуально-орієнтований інтерфейс був розроблений для навчання школярів основам програмування та обчислювального мислення. Однак, його застосування не обмежується лише цією групою користувачів, і воно також знайшло широке застосування в галузі робототехніки.

Аналіз стану навчання робототехніки в закладах освіти показав, що її вивчення здійснюється: окремими темами на уроках інформатики або технологій у закладах загальної середньої освіти; за спеціально розробленими програмами гурткових занять у позашкільних закладах за інтересами дітей з урахуванням вікової категорії; окремою спеціальністю чи дисциплінами «Освітня робототехніка», «Робототехнічні системи», «Stem-освіта», «Штучний інтелект» та ін. у закладах вищої освіти. Як бачимо, робототехніка є універсальним інструментом для освіти, вона доцільно вписується у вищу освіту, в позашкільне навчання, у навчальні предмети шкільної програми. Крім того, робототехніка може бути використана для інтеграції з іншими дисциплінами навчального

плану, створюючи міжпредметні зв'язки та сприяючи глибшому розумінню матеріалу. Наприклад, здобувачі можуть створювати роботів, які вивчають екосистеми в рамках біологічних наук або роботів, що моделюють фізичні закони.

Спостерігаємо масове вивчення робототехніки у гуртках інформаційно комп'ютерного профілю, вихованцями яких є діти віком від 10 до 15 років. Складність занять зумовлена використанням технічного і програмного забезпечення. На ринку робототехніки є різні платформи, спеціально розроблені для впровадження в освітній процес. Так, у закладах загальної і позашкільної освіти використовують робототехнічні набори на основі Arduino, конструктори і платформи від компанії Lego. Навчання спрямоване на складання роботів, кожен з яких ототожнюється із механічним пристроєм, який запрограмований для виконання інструкцій (Гойда & Павлова, 2021, с. 96).

Scratch є найбільшою в світі безкоштовною програмою для школярів, яка дозволяє створювати власні інтерактивні історії, ігри та анімації. Це середовище програмування розроблене некомерційною організацією Scratch Foundation.

Програмування в Scratch включає переміщення блоків, подібних до пазлів, що дозволяє створювати зображення з унікальними відповідями, вставляти аудіокліпи та звукові ефекти, щоб урізноманітнити проект (Papert, 1980).

Scratch також надає можливість завантажити безкоштовний додаток, який дозволяє створювати та зберігати проекти без підключення до Інтернету. Це робить Scratch доступним для користувачів, які мають обмежений доступ до Інтернету.

Scratch надає учням можливість вчитися, експериментувати та творити в безпечному та підтримуючому середовищі. Це сприяє розвитку критичного мислення, творчих навичок.

Мова Scratch оперує числами, текстовими рядками, логічними значеннями, а також списками, що відіграють роль динамічних масивів. Scratch добре підходить для навчання, моделювання, керування пристроями. Для створення програмних проектів Scratch має всі необхідні засоби: мову програмування,

інтерпретатор мови, графічний редактор, систему допомоги, зразки проектів, бібліотеку малюнків та звукових файлів (Гойда & Павлова, 2021).

Дослідження показують, що використання Scratch сприяє розвитку в учнів також і обчислювального мислення. Вони навчаються розуміти логіку програм та алгоритмів, вивчають умовні оператори та цикли, а також набувають навичок у роботі зі змінними та подіями. Scratch також сприяє розвитку творчого мислення учнів. Вони можуть створювати власні проекти, ігри, анімації та інтерактивні історії. Це стимулює їхню уяву та допомагає розвивати навички проблемного мислення. Використання цієї програми в освітньому процесі сприяє розширенню можливостей навчання робототехніки та підготовці школярів до вимог сучасного цифрового світу (Anderson & Anderson, 2010, с. 293-299).

У світі освіти STEM (Science, Technology, Engineering, and Mathematics) і робототехніки дуже важливо мати доступ до засобів, які сприяють навчанню та розвитку цих галузей. Spike Prime є освітньою платформою від компанії LEGO Education, що створена для навчання STEM та робототехніки у школах.

Робототехніка та STEM-освіта стають все більш важливими в сучасному світі, оскільки вони розвивають навички критичного мислення, творчості та технологічної грамотності. Spike Prime, розроблений LEGO Education, поєднує у собі навчальні ігри, конструктори та програмне забезпечення для навчання STEM-предметів та робототехніки. конструктор є провідним засобом навчання освітньої робототехніки, що представлений набором деталей, сенсорів, з'єднувальних елементів, із яких можна зібрати різні види роботів. Конструктор LEGO: Spike Prime (рис. 1.3.) використовує деталі LEGO, які дозволяють учням будувати різноманітні конструкції та роботи. Це сприяє розвитку творчості, просторої уяви, мисленнєвій активності.

Також конструктор Lego дозволяє учням у формі пізнавальної гри поглибити інтерес про робототехніку та розвинути навички. Lego-робот допомагає зрозуміти основи робототехніки, наочно реалізувати складні алгоритми, проаналізувати питання, пов'язані з автоматизацією виробничих

процесів та процесів управління. Середовище програмування LEGO: Spike Prime можна джзавантажити як на комп'ютер, так і на планшет або на телефон

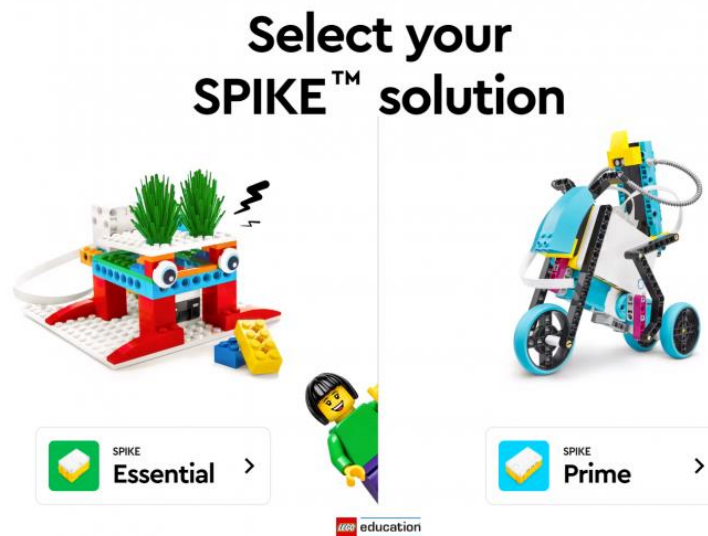


Рис. 1.3. Вигляд програми Spike Prime

Основні характеристики Spike Prime:

1. **Сенсори та актуатори:** платформа обладнана різними сенсорами, які дозволяють роботам взаємодіяти з навколишнім середовищем. Вона також включає мотори для виконання різних завдань.
2. **Програмне забезпечення:** Spike Prime має вбудоване програмне забезпечення з графічним інтерфейсом, яке дозволяє учням програмувати роботів. Вони можуть створювати складні алгоритми та керувати рухами роботів.
3. **Навчальні ресурси:** LEGO Education надає вчителям навчальні матеріали та уроки для використання Spike Prime у класах. Це допомагає інтегрувати платформу в шкільну програму.
4. **Інтеграція зі шкільною програмою:** Spike Prime розроблено з урахуванням навчальних стандартів, що сприяє його успішній інтеграції в освітній процес.

Spike Prime допомагає розвивати навички технологічної грамотності, творчості, критичного й обчислювального мислення. Він сприяє інтерактивному

та практичному навчанню, стимулюючи інтерес до науки та технології серед учнів. Використання Spike Prime в класі дозволяє учням досліджувати STEM-предмети в забавній та зрозумілій формі.

Spike Prime – це інноваційна навчальна платформа, яка сприяє розвитку STEM-навчання та робототехніки в освітніх закладах. Вона надає можливість учням розвивати навички мислення і технологічної грамотності через практичне навчання та творчість.

1.3. Принципи роботи Scratch та Spike Prime

Щоби розпочати створення власної гри або анімації, школярам потрібно спершу вивчити інтерфейс програми. Гарна новина в тому, що він є дуже простим та інтуїтивно зрозумілим. Він має чотири основних елементи (рис. 1.4.):

1. Зона сцени – місце для відображення результатів.
2. Спрайт – головний герой проєкту й об'єкт, що, на відміну від зони сцени, може пересуватися та змінюватися. У програмі Скретч та на сторонніх сайтах існує багато пакетів зі спрайтами, розбитих за тематиками, наприклад, автомобілі чи тварини. Спрайти також можуть змінювати зовнішній вигляд за допомоги костюмів.
3. Палітра блоків – кількість блоків із командами, що знаходяться в лівій частині екрана.
4. Область кодування – містить блоки, котрі перетягуються з палітри Блоків. Тут зберігається та редагується вже написаний код проєкту (Introduction to Scratch Programming Language [Електронний ресурс]. 2023.).

Щоб створити перший проєкт у Scratch, потрібно спочатку вивчити область кодування та її можливості. Крім того, корисним буде розпочати використання Scratch з вивчення інших проєктів та розробок. Це допоможе усвідомити можливості цієї мови програмування. Далі можна копіювати проєкт, розроблений іншими користувачами та змінювати його відповідно до власного задуму. Після цього можна спробувати створити власний проєкт. Для цього ви можете запропонувати учневі зробити аналог її улюбленої відеогри.

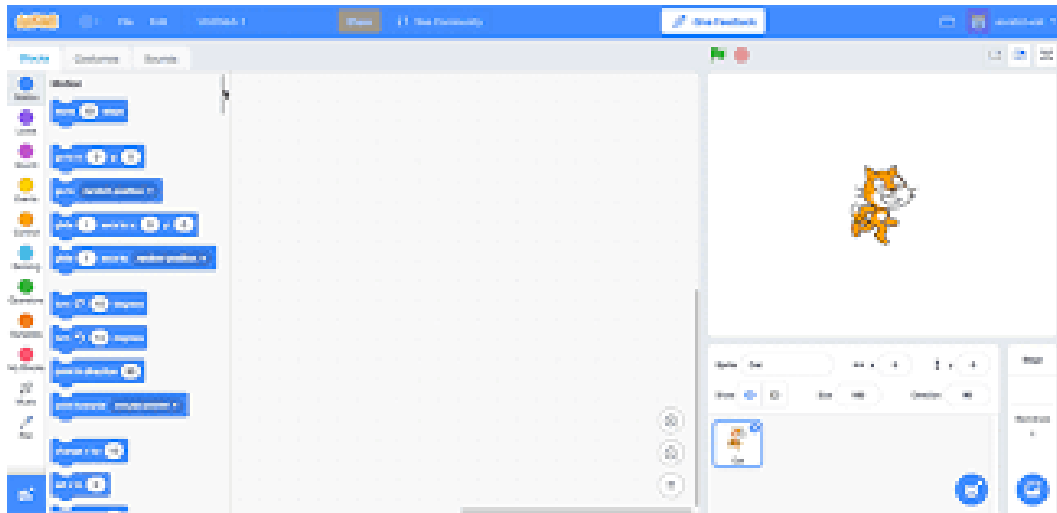


Рис. 1.4. Інтерфейс програми Scratch

Важливими етапами для опанування Scratch буде вивчення того, як створювати функції, управляти об'єктами або спрайтами, як організувати взаємодію між об'єктами, як клонувати об'єкти та управляти клонами. Крім того, учневі може навчитися симулювати гравітацію, працювати зі списками, управляти кольорами та звуковими ефектами.

Разом із фахівцем учень зможе не лише навчитися основ, а й створити свій проєкт із застосуванням максимальної кількості функцій цієї мови.

А тепер про Spike Prime. Першим кроком, коли потрібно написати програму, є запуск програми Spike. Щоб написати нову програму, натиснути кнопку «Новий проєкт» посередині екрана. Програма має більше функціональних можливостей від вище згаданої мови програмування, оскільки в даному наборі міститься більша кількість датчиків і до кожного із них потрібні окремі блоки.

Далі необхідно вибрати, який тип програми створити. Оскільки, що є кілька варіантів. Блоки значків повинні бути корисними для дітей, які навіть не вміють читати. Можна почати з Word Blocks у тому випадку, коли учні знайомі зі Scratch, оскільки програма має подібний функціонал. Програма також володіє можливостями Python. Для початку варто натиснути «word blocks» (рис. 1.5). Завжди можна спробувати простішу версію піктограм або більш просунуту

версію Python пізніше. Вицвілі літери показують, що програму буде названо Project 1. Є можливість назвати проект.

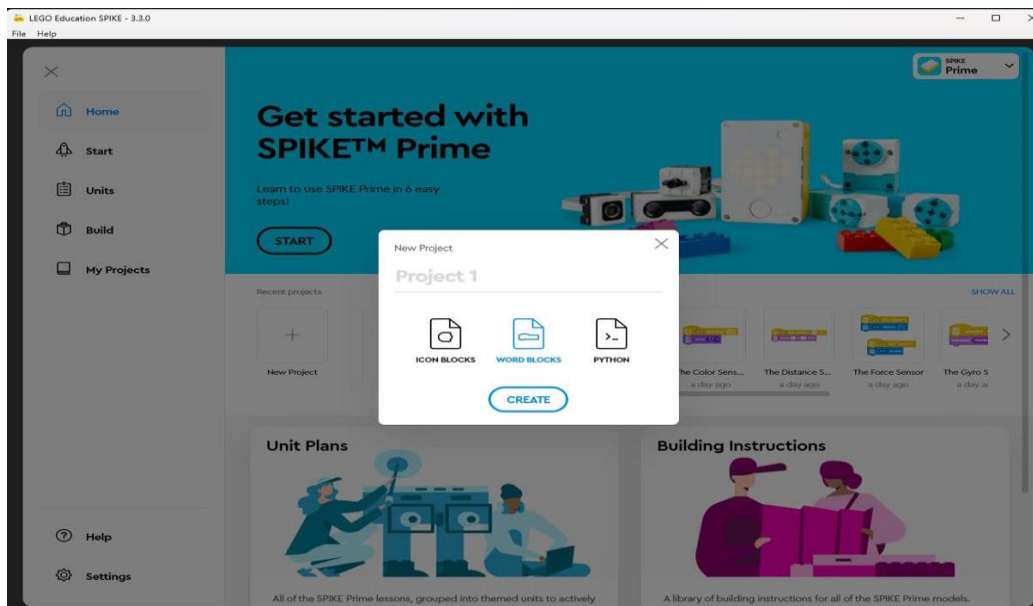


Рис. 1.5. Створення проекту в Spike Prime

Якщо коротко описати роботу у даній програмі то спочатку потрібно зібрати робота з конструктора LEGO, а потім запрограмувати Spike Prime. У послідовності дій виокремимо:

1. Підключити сенсори: підключити різні сенсори, які допоможуть роботу "бачити" оточуюче середовище і взаємодіяти з ним. Наприклад, сенсор відстані або сенсор кольору.
2. Запрограмувати робота: використати програмне забезпечення Spike Prime, яке має графічний інтерфейс, щоб створити програми для робота. Користувач можете вказувати, як робот має рухатися, реагувати на сенсори та виконувати різні завдання.
3. Випробувати і вдосконалити: запустити програму на роботі, надати йому можливість діяти згідно програми; за потреби відкоригувати програму.
4. Масштабність: складати робота і використовувати Spike Prime для вивчення робототехніки та створення власних проектів.

1.4. Типи мислення у школярів

Наукові дослідження в галузі психології та освіти вказують на різноманітні типи мислення у школярів, що визначають їхні здібності та підхід до навчання. Розглянемо деякі з них на основі наявної літератури:

- **Критичне мислення:** є важливим елементом освітнього процесу, спрямованим на розвиток аналітичних навичок та здатності оцінювати інформацію (Пауль & Елдер, 2006). Воно дозволяє учням не лише сприймати факти, але й критично їх аналізувати та робити висновки. Роль критичного мислення полягає у тому, щоб протистояти вірі, що не піддається поясненню. Цей вид мислення проявляється у двох напрямках: 1) орієнтація на пошук правди; 2) розуміння різноманітності. Віра в абсолютизм і в ідеї, які не піддаються поясненню, є негативними показниками критичного мислення (Терно, 2012, с. 70).

- **Творче мислення:** визначається як здатність генерувати нові ідеї, виконувати діяльність, що створює якісно нові, унікальні інформаційні об'єкти, формує нестандартні рішення проблем. В шкільному середовищі це може виявлятися у здатності до нестандартного підходу до завдань і вираженні оригінальних думок.

- **Логічне мислення:** орієнтоване на розвиток систематичної та послідовної послідовності дій, індуктивних (від фактів до гіпотези, тобто від часткового до загального) і дедуктивних (від загального до конкретного) міркувань. Учні з цим типом мислення виявляють здатність до визначення логічних зв'язків між ідеями, логічних наслідків та їх структурування.

- **Візуальне мислення:** іншим типом мислення є візуальне мислення, яке проявляється у здатності до розуміння та використання візуальних зображень для сприйняття інформації (Рензулі, 2012). Це може бути корисним у навчанні, де важлива візуальна репрезентація концепцій.

Актуальним для дослідження є **алгоритмічне мислення**, яке зіставляємо: 1) із розумовими способами дій, прийомів для вирішення задач; 2) із створенням алгоритму; 3) із переходом від безпосереднього управління до програмного; 4) із

пізнанням, що характеризується наявністю чіткої, доцільної послідовності розумових процесів. У загальному алгоритмічне мислення є сукупністю розумових дій, прийомів, результатом застосування яких є алгоритми.

Поняття алгоритмічного мислення ширше, ніж кожне згадане вище мислення». Це зумовлене тим, що алгоритмічне мислення передбачає розуміння змісту базових алгоритмічних конструкцій, (слідування, розгалуження, цикл), а також уміння їх використовувати при складанні алгоритмів.

З іншого боку, згадані типи мислення у школярів взаємодіють та визначають індивідуальний підхід до навчання, а розвиток кожного з них може бути сприятливим завдяки відповідним педагогічним стратегіям. Розуміння цих типів мислення дозволяє ефективно структурувати програми та підтримувати розвиток учнів у різних аспектах їхнього когнітивного потенціалу (Козир & Грабовська, 2019, с. 8).

Систематизуючи сутність алгоритмічного, логічного, творчого мислення виділимо вміння які можна сформулювати, адаптуючи їх до особливостей робототехніки:

- структурний аналіз задачі через розбиття її на підзадачі та оперування образами і поняттями;
- моделювання послідовності дій, яка необхідна для вирішення кожної підзадачі, а далі синтез знайдених рішень в один результат;
- пошук ідеї розв'язування, розглядаючи усе завдання з подальшою деталізацією відповідно до підзадач;
- декомпозиція задачі на рівні процесів;
- формалізація задачі через індуктивні і дедуктивні висновки (впорядкування операцій, побудова моделі розв'язування в тому числі і графічний спосіб);
- аналогія алгоритм розв'язування задачі у середовищі програмування.

Різні типи мислення можуть впливати на те, як людина підходить до структури та організації коду, зокрема до використання циклів та розгалужень у програмуванні.

Найважливішим завданням сучасної системи освіти є формування універсальних навчальних дій, завдяки яким школяр навчиться вчитися, набуде здатності до саморозвитку та самовдосконалення. Усе це досягається шляхом свідомого, активного набуття учнями соціального досвіду. При цьому знання, вміння та навички розглядаються як похідні від відповідних видів цілеспрямованих дій. Вони формуються, застосовуються та розвиваються у поєднанні з активними діями самих учнів. Такі активні дії у початкових класах можуть бути забезпечені шляхом застосування технологій освітньої робототехніки. Навчання у цей період має сприяти розвитку в учнів мотивації подальшої освітньої діяльності, і навіть інформаційної культури, як уміння цілеспрямовано опрацювати інформацією та використовуватиме її (Конотоп & Матвієнко, 2023, с. 715).

Програмування в середовищі Scratch включає в себе використання циклів та розгалужень для керування виконанням програм. Ці конструкції дозволяють створювати більш складні та ефективні алгоритми, роблячи програму більш гнучкою та динамічною.

У мові програмування Scratch для реалізації циклів використовується блок «повторити». Цей блок дозволяє визначити кількість повторень операцій, що мають бути виконані. Наприклад, блок:

У мові програмування Scratch для реалізації циклів використовується блок «Повторити». Цей блок дозволяє визначити кількість повторень операцій, що мають бути виконані. Наприклад, блок (рис. 1.6.) повторить вкладені операції п'ять разів після чого спрайт переміститься на 50 кроків (Scratch [Електронний ресурс], 2023).

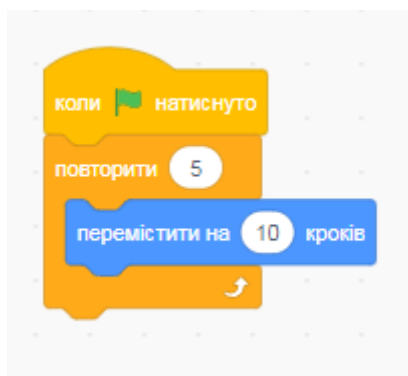


Рис. 1.6. Цикл «повторити» у програмі Sctach

У Scratch існує блок, який називається "Завжди", і це особливий вид циклу, який забезпечує постійне виконання певних операцій під час роботи програми. Цей блок дозволяє вкладати в себе операції, які будуть виконуватися безперервно, не очікуючи на виконання певної умови. Використання блоку «завжди» може виглядати як на рис. 1.7.

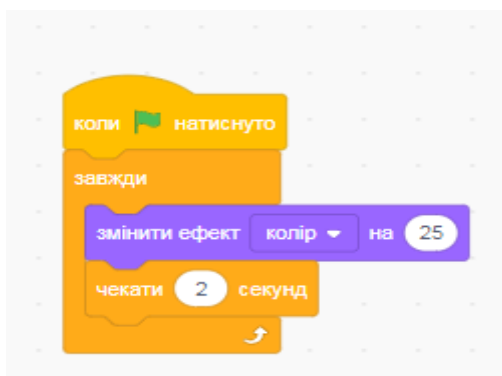


Рис. 1.7. Цикл «завжди» у програмі Scratch

Цей блок може містити будь-яку кількість операцій, і вони будуть виконуватися неперервно, що робить його корисним для ситуацій, де операції повинні відбуватися без зупинки. Блок «Завжди» часто використовується для виконання постійних функцій або взаємодії з користувачем, що робить його корисним у веб-проектах, іграх або інших програмах, де деякі операції повинні відбуватися без перерви. Або у програмі також можна використовувати розгалуження. Розгалуження – це структура управління, яка дозволяє виконувати різні команди залежно від результату виконання перевірки. У Scratch існує два типи розгалужень: повне (якщо-то-інакше) - виконує одну з двох груп

команд залежно від результату перевірки; неповне (якщо-то) - виконує групу команд, якщо результат перевірки є істинним.

Наприклад, наступний код (рис. 1.8.) буде виконати команду «Виконавець говорить «Привіт!», якщо виконавець торкнеться вказівником спрайта, і команду «Виконавець говорить «До побачення!», якщо виконавець не торкнеться спрайта. Даний код програми є прикладом повного розгалуження. Або якщо брати неповне розгалуження, то код програми буде приймати наступний вигляд. Програма буде працювати лише за умови коли виконавець буде торкатися вказівником спрайта. Узагальнюючи, цикли та розгалуження у Scratch є потужними інструментами для створення програм, які вирішують різноманітні завдання, включаючи наукові дослідження. Їх правильне використання дозволяє створювати ефективні та структуровані програми.

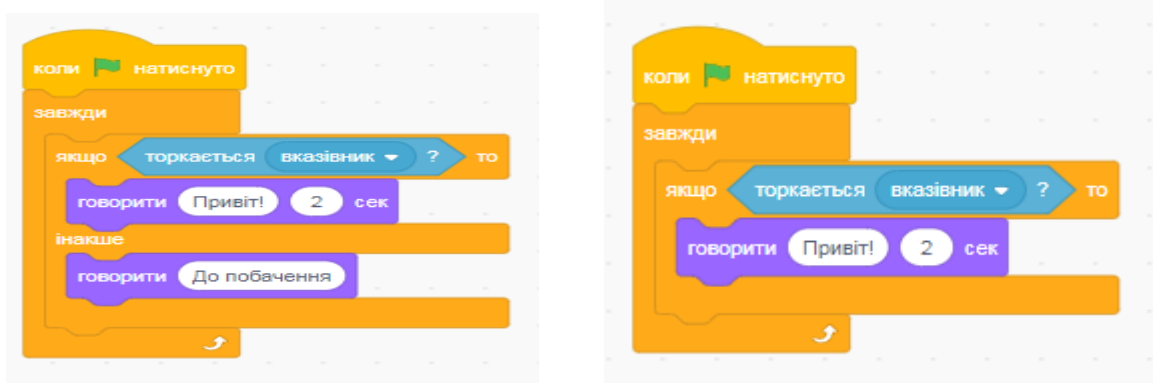


Рис. 1. 8. Приклад повного і неповного розгалуження

РОЗДІЛ 2. SCRATCH ДЛЯ РОБОТОТЕХНІКИ

2.1 Історія та основні концепції Scratch

Scratch – широко розповсюджена мова візуального програмування, розроблена Lifelong Kindergarten Group медіа-лабораторії МІТ. Вона має захоплюючу історію та зазнала кількох винаходів та ітерацій з моменту створення. Ось короткий огляд історії та ключових винаходів Scratch:

Scratch Emerges (2003): Scratch спочатку був випущений у 2003 році як дослідницький проект Lifelong Kindergarten Group Массачусетського технологічного інституту. Він був створений Мітчелом Резніком та його командою, щоб забезпечити креативну та доступну платформу для молодих учнів для вивчення концепцій програмування.

Ключові винаходи та відповідні їм роки:

Блокове програмування (2003): одним із ключових винаходів Scratch була концепція блокового програмування. Замість того, щоб вводити рядки коду, користувачі Scratch перетягують і з'єднують різнокольорові блоки коду, роблячи його доступним для користувачів із незначним досвідом кодування або без нього.

Інтернет-спільнота (2006): У 2006 році було запущено онлайн-спільноту Scratch. Ця платформа дозволяла користувачам ділитися своїми проектами, співпрацювати та вчитися один у одного. Це відіграло ключову роль у популярності та освітньому впливі Scratch.

Scratch 2.0 (2013): у 2013 році був випущений Scratch 2.0, який значно покращив платформу. Це дозволяло користувачам створювати проекти безпосередньо у своїх веб-браузерах, роблячи Scratch ще доступнішим.

Розширення до апаратного забезпечення (2014): Scratch вийшов за межі просто програмного забезпечення та почав підтримувати інтеграцію з апаратним забезпеченням, включаючи популярний комплект робототехніки LEGO Mindstorms EV3. Цей крок заохочував фізичні обчислення та робототехніку.

Scratch 3.0 (2019): Scratch 3.0 представив оновлений і більш зручний інтерфейс. Він також включав підтримку нового апаратного забезпечення, такого як micro:bit і Makey Makey, що ще більше розширило його охоплення.

Іншими основними особливостями Scratch є:

1) блочне програмування. Для створення програм потрібно поєднати графічні блоки. Блоки зроблені так, що їх можна зібрати лише в стеках в синтаксично вірних конструкціях, уникаючи таким чином можливих помилок на етапі розробки програми. Різні типи даних мають різні форми, підкреслюючи несумісність окремих типів. Наявність такої характеристики, як динамічність дозволяє розробнику змінювати код навіть під час виконання програми, таким чином з'являється більше можливостей для експериментування з новими ідеями;

2) маніпуляції з даними. Scratch дозволяє користувачу створювати програми, які поєднують графіку, анімацію, музику та звуки. Scratch розширює можливості управління візуальними даними, використання яких в сьогоdnішній інформаційній культурі є популярним.

Учням, які усвідомили функціонування програми та засвоїли початкові навички програмування на певному етапі роботи пропонується програмувати роботів. Для виконання такої діяльності середовище Scratch має безліч додаткових функцій, зокрема додавання інших частин, однією з яких є SmartHub.

SmartHub WeDo 2.0 є процесором, який є об'єктом програмування, керуючи датчиками і двигунами. Він виконаний в стилі «розумної цеглини» EV3, але істотно меншого розміру. Блок не дозволяє програмувати на ньому самому, має одну кнопку, яка відповідає за включення і запуск пошуку блоку комп'ютером чи планшетом згідно протоколу Bluetooth 4.0.

Програмування здійснюється за допомогою комп'ютера або планшета після підключення. Картки програмування Scratch (2020): у 2020 році Массачусетський технологічний інститут випустив набір карток програмування Scratch, які містили фізичні картки з блоками Scratch і діями для дослідження програмування в автономному режимі (Гойда, Павлова, 2021).

Scratch став цінним інструментом для школярів у всьому світі. Його використовують у класах для навчання кодуванню, вирішенню проблем і творчому мисленню. Scratch також присутній у неформальних навчальних закладах, таких як клуби кодування та семінари. Вплив Scratch поширюється по всьому світу, завдяки різноманітній базі користувачів, які представляють різні культури та мови. Функції перекладу платформи роблять її доступною для користувачів із різним становищем. Scratch надихнув користувачів на створення широкого спектру проектів, включаючи анімацію, ігри, інтерактивні історії та симуляції. Ця креативність сприяла привабливості Scratch як засобу навчання (Introduction to Scratch Programming Language [Електронний ресурс]. 2023).

2.2. Використання Scratch для програмування роботів

Для програмування сценаріїв у Scratch використовується drag-and-drop-підхід: блоки з панелі блоків перетягуються в область скриптів. За функціональним призначенням блоки діляться на 10 груп, приналежність блоку до тієї чи іншої групи позначається його кольором (таблиця Б.1, додаток Б).

Однією з головних концепцій Scratch, є розвиток власних ідей у вигляді алгоритму до програмного продукту. Для цього в Scratch є необхідні засоби:

- стандартні: слідування, розгалуження, цикли, а також змінні, типи даних (цілі та дійсні числа, рядки, логічний тип, списки), псевдовипадкові числа;
- об'єктно-орієнтовані: об'єкти (їх поля і методи), передача повідомлень і обробка подій;
- інтерактивні: обробка взаємодії об'єктів між собою, з користувачем, а також подій поза ПК (підключення сенсорного блоку);
- паралельне виконання: запуск методів об'єктів в паралельних потоках з можливістю координації і синхронізації;
- створення простого інтерфейсу користувача.

Програмувати LEGO Education WeDo 2.0 у Scratch 3.0 або з'єднати два феноменальні інструменти для навчання робототехніки та програмування.

LEGO Education WeDo 2.0 (рис. 2.1) – один із найкращих інструментів для учнів, які хочуть вивчати робототехніку. Набір містить спеціально підібрані кубики LEGO та основні електронні елементи: один двигун, два датчики та концентратор для підключення вашої конструкції до комп'ютера чи планшета.



Рис. 2.1. Набір Lego Education WeDo 2.0

У програмному забезпеченні для програмування, призначеному для WeDo 2.0, учні можуть відкрити для себе основні команди програмування, але загальні можливості обмежені. Комплекти WeDo 2.0 можна програмувати на добре відомій і улюбленій мові програмування Scratch, яку цінують як діти, так і дорослі вчителі.

Разом зі Scratch і WeDo школярі почнуть вивчати основи програмування, а також буде можливість побачити результат на власні очі. Вони також відкривають елементи інженерії та STEM.

WeDo і Scratch є перевіреними навчальними інструментами. Кожен має кілька попередніх версій, а це означає, що розробники кожного продукту мали можливість виправити помилки та вдосконалити. Згодом починати для новачків стало простіше, можливостей не зменшилося, а найнадійніші рецензенти – діти

– досі люблять ці інструменти. Знайти будь-які недоліки в найновіших версіях, наприклад, Scratch 3.0 і LEGO WeDo 2.0, є викликом.

Щоб запрограмувати WeDo в Scratch, потрібні 3 компоненти:

- Набір LEGO Education WeDo 2.0;
- Програма Scratch 3.0;
- Пристрій для програмування (наприклад, ноутбук або планшет).

Щоб відкрити Scratch, достатньо ввести у браузері scratch.mit.edu, але в стандартній палітрі кодових блоків не можливо знайти команди, які дозволяють керувати двигуном або датчиками з WeDo 2.0. Щоб їх знайти потрібно виконати наступні кроки:

1. Відкрити головну сторінку Scratch, далі перейти до редактора. Натиснути значок глобуса у верхньому лівому куті та обрати мову.

2. Підготувати інтерфейс. Під категоріями блоків знайти синій значок «Додати розширення» (рис. 2.2) та натиснути на нього. Далі обрати розширення під назвою LEGO Education WeDo 2.0 (він рожевий).

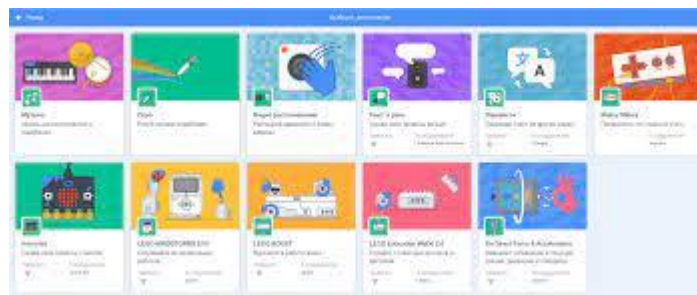


Рис. 2.2. Розширення програми Scratch

3. Встановити програму Scratch Link на свій пристрій. Скачати з офіційного сайту відповідний інсталятор для вашої ОС і запустити його. Scratch Link дає змогу підключити WeDo 2.0 Smarthub до Scratch, тому він завжди повинен працювати у фоновому режимі, поки створюється програма.

4. Підключити Smarthub до Scratch. Увімкнути підключення Bluetooth на пристрої. У Scratch клацнути піктограму зі знаком оклику, щоб відкрити вікно пошуку Smarthub. Натискати зелену кнопку на Smarthub, доки світлодіод не

почне блимати, а потім натиснути «Почати пошук» у спливаючому вікні Scratch. Дочекатися повідомлення про підтвердження встановлення підключення.

5. Повернутися до редактора. Активувати кілька блоків із категорії блоків WeDo 2.0, щоб перевірити, як працюють двигун і датчики, підключені до Smarthub.

2.3. Як скласти робота засобами роботехнічного конструктора

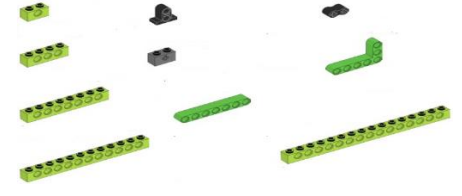


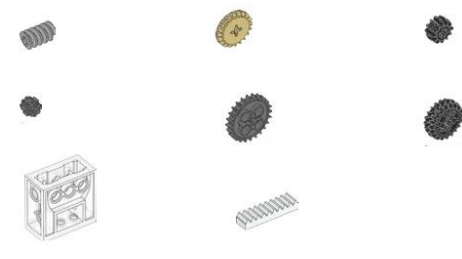
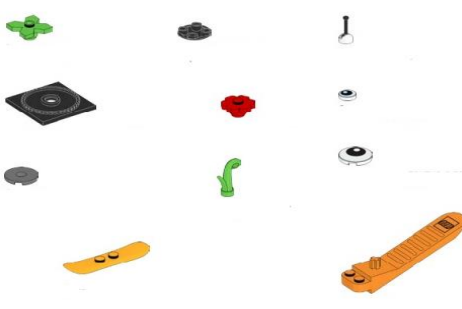


Блоковий конструктор Lego Education WeDo 2.0 (рис. 2.3) є класичним конструктором, з якого учні можуть зібрати різні моделі робота. 280 деталей (табл. 2.1), серед яких Lego Smart Hub 2.0, електромотор, датчики нахилу і руху дозволяють з'єднувати елементи різної конфігурації у конструкцію робота. Програмне забезпечення WeDo 2.0, сумісне з різними версіями операційних систем, дозволяє роботу рухатися. Комплект містить навчальні матеріали для реалізації 17 проектів з навколишнього світу, біології, географії, дослідження космосу та інженерного проектування, робота над якими загалом може зайняти понад 40 академічних годин. Мікрокомп'ютер використовує протокол Bluetooth 4.0 для з'єднання з комп'ютером або планшетом. Робота з таким конструктором дозволяє розвивати в учнів навички конструювання робота, проектування алгоритму дій та програмування. Комплектацію набору Lego Education WeDo 2.0 представлено в табл. 2.1. В ній показано основні (найчастіше використовуванні учнями) деталі для конструювання роботів.



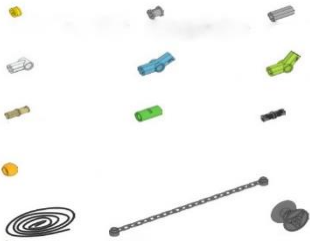
Рис. 2.3. Блоковий конструктор Lego Education WeDo 2.0

Таблиця 2.1

Склад конструктора Lego Education WeDo 2.0

Вигляд компонентів	Характеристика компонентів
	<p>Балки – використовуються для створення міцного каркасу конструкції; розрізняють технічні та звичайні балки;</p>
	<p>Вісі – передають оберти від мотора колесам, шестерням; використовуються для створення шасі або зубчатих та ремінних передач;</p>
	<p>Конектори – слугують з'єднаннями для балок, датчиків.</p>
	<p>Шестерні призначені для передачі обертання від моторів до інших елементів конструкції робота. Тут можна познайомитися із коробкою для черв'ячної передачі. Дана деталь допомагає учням у побудуванні черв'ячної передачі.</p>
	<p>Декоративні деталі використовують для надання оригінальності зовнішньому вигляду робота. Школярам подобається прикрашати робота, таким чином, щоб він мав певну особливість і відрізнявся від інших.</p>
	<p>Скоси та цеглинки LEGO мають подібну функцію, як і балки Вони зміцнюють каркас моделі, щоб робот під час запуску програми «залишився цілим».</p>
	<p>Пластини – скріплюючі деталі</p>

Продовження таблиці 2.1

	<p>Кутові блоки використовують для з'єднання осей під кутом або для того, щоб їх подовжити. Також в наборі є «котушка», яка призначена для демонстрації принципу роботи лебідки.</p>
	<p>LEGO Education WeDo 2.0 Smart Hub – це бездротовий з'єднувач між комп'ютером або планшетом і датчиками або мотором WeDo, що використовує технологію Bluetooth.</p>

LEGO Smart Hub – «мозок» будь-якої зібраної за інструкцією або власноруч згенерованої моделі, використовуючи технології LEGO Education або Mindstorms EV3. Основна функція LEGO Smart Hub – це бездротовий зв'язок Bluetooth з комп'ютером, смартфоном або планшетом, які містять програмне забезпечення. Він отримує команди програми від пристрою та виконує їх. LEGO Smart Hub має два порти для підключення датчиків або двигунів, індикатор і кнопку живлення.

WeDo 2.0 LEGO Education (рис. 2.4) допомагає «оживити» конструкцію та додати їй нові ідеї. LEGO-робот з двигуном WeDo 2.0, може не тільки рухатися, але і змінювати напрямок руху і регулювати швидкість за допомогою таймера у двигуні, задаючи час руху з подієвою точністю. На пристрої є кріплення для інших цеглинок LEGO, тому двигун є частиною конструкції і не змінює її зовнішній вигляд. Розпізнавання пристрою програмним забезпеченням відбувається автоматично. Розширюють можливості робота датчик нахилу і датчик руху, останній з яких дозволяє роботу виявити об'єкти в діапазоні 15 см. і тому використовується для виявлення змін у відстані до об'єкта трьома способами: наближення, віддалення, зміна положення.



Рис. 2.4. Двигун та датчики нахилу та руху WeDo 2.0

Поєднуючи елементи Lego у складний механізм, учні розвивають логічне мислення, просторову уяву, навчаються програмувати, задаючи умови руху для двигуна. За таких умов учні навчаються, розвиваються, досягають нові знання у поєднанні з сучасними технологіями.

2.4. Приклади проектів з використанням Scratch

Після розгляду можливості програми Scratch та конструктора Lego Education WeDo 2.0 розглянемо прикладами моделей, які можливо сконструювати та запрограмувати.

Перша з них являє собою модель вантажівки (рис. 2.5) з використанням Scratch для програмування використовує різні механізми для своєї роботи:

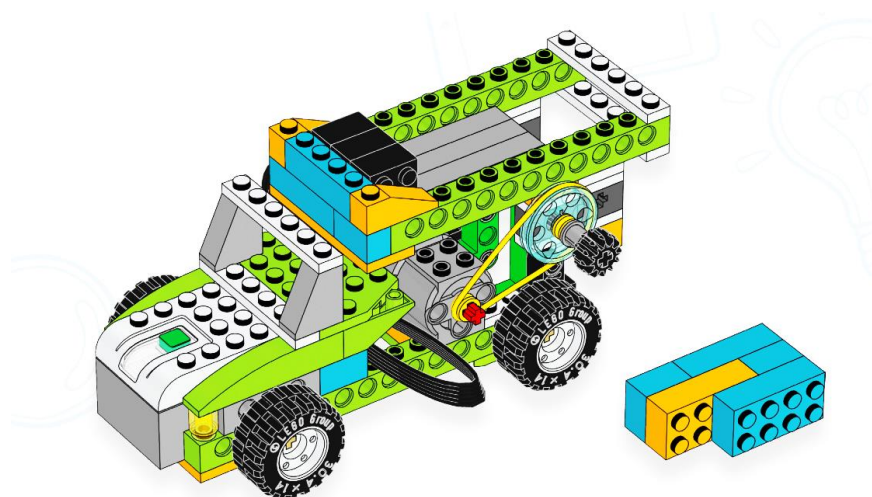


Рис. 2.5. Модель: «Вантажівка»

- Зубчаста передача використовується для приведення у рух вантажівки. Зубчаста передача може забезпечувати різну швидкість та потужність руху, в залежності від того, як вона налаштована.

- Ремінна передача використовується для піднімання кузова вантажівки. Ремінна передача може бути використана для перенесення руху з одного місця на інше, в даному випадку для піднімання або опускання кузова вантажівки.

Це хороший приклад використання механічних систем у робототехніці та програмуванні за допомогою Scratch. Модель вантажівки дозволяє учням та іншим зацікавленим особам вивчати основи механіки та програмування в забавний та інтерактивний спосіб.

Наступним побачимо «Алігатора» (рис. 2.6). Використовуючи програмне забезпечення WeDo 2.0, діти будують голодного алігатора. Потім вони програмують «Алігатора» відкривати та закривати рот за допомогою власних алгоритмів, а далі дітям пропонується використовувати датчик руху, щоб контролювати звіра, що тріскається.



Рис. 2.6. Модель: «Голодний алігатор»

2.5. Переваги та недоліки програми Scratch

Хоча програма Scratch дає необхідні знання для того, щоб ознайомитися із основами програмування, вона має свої плюси та мінуси. Переваги програми Scratch:

1. **Візуальне програмування:** Scratch використовує блоки програмування, які можна перетягувати та з'єднувати, що робить програмування

доступним і зрозумілим для початківців. Візуальний підхід полегшує створення програм.

2. **Безкоштовність та доступність:** Scratch є безкоштовною програмою, є можливість використовувати її онлайн без необхідності завантаження або встановлення на компютер користувача. Це робить її доступною для багатьох користувачів.
3. **Громадська спільнота:** Scratch має активну громадську спільноту користувачів і розробників. Розробники середовища розмістили у програмі та в інтернеті безліч проектів, навчальних матеріалів.
4. **Навчання програмування:** Scratch допомагає вчити основи програмування, включаючи роботу зі змінними, умовними операторами, циклами та подіями. Що урізноманітнює вивчення робототехніки та є пропедевтикою вивчення програмування, розвиваючи як алгоритмічне мислення, так і інші його види.
5. **Можливість створення різних проектів:** Scratch має можливість створювати ігри, анімації, інтерактивні історії та багато інших проектів в Scratch, що сприяє творчості та розвитку навичок.

Далі розглянемо деякі пункти детальніше. Візуальне програмування - це підхід до створення програм, який дозволяє програмістам використовувати графічні компоненти для побудови логіки програми. Замість писання коду вручну, програмісти перетягують і з'єднують блоки або символи, що представляють функції та операції. Візуальне програмування використовує блоки, які представляють різні операції, умови, цикли та інші елементи програми. Вони можуть бути перетягнуті на полотно і з'єднані для створення програмної логіки. У програмі легко визначати логічні зв'язки між блоками. Наприклад, створити умови "якщо-інакше" або цикли "повторити". Або створити блоки, які реагують на різні події, такі як клік мишею або натискання клавіші.

Візуальне програмування - це інструмент для створення програм, особливо для початківців і тих, хто не володіє великими навичками програмування. Воно спрощує процес створення програм та полегшує їх розуміння.

Громадська спільнота у Scratch - це група людей, які користуються цією програмою для створення і розповсюдження своїми проектами та ідеями. Учасники громадської спільноти можуть створювати різні проекти, такі як ігри, анімації, інтерактивні історії та багато інших речей, використовуючи Scratch. Потім вони можуть ділитися цими проектами з іншими користувачами, щоб показати свою творчість. Користувачі можуть залишати коментарі на проекти інших користувачів, висловлювати свою думку та залишати рекомендації. Це сприяє обміну досвідом та навчанню. Scratch має форуми, де користувачі можуть обговорювати різні теми, задавати питання, шукати допомогу та спілкуватися з іншими учасниками спільноти. Учасники спільноти можуть брати участь в різних спільних завданнях та змаганнях, які організовані на платформі Scratch. Це може бути стимулом для творчості та розвитку навичок. Громадська спільнота також надає навчальні матеріали та уроки для тих, хто бажає вивчити більше про Scratch та програмування.

Усе це робить громадську спільноту в Scratch дружним та інформативним місцем, де користувачі можуть навчатися та надихатися іншими для створення цікавих та творчих проектів. Мінуси програми Scratch:

1. **Обмежена складність проектів:** Scratch, хоч і дозволяє створювати багато різних проектів, має свої обмеження в складних програмах та обмежений у своїх ресурсах. Він може бути недостатнім для більш серйозних програмістів.
2. **Відсутність розширених програмувальних мов:** Scratch використовує свою власну блокову мову програмування, і він не підтримує більш розширені мови, такі як Python або JavaScript.
3. **Можливість обмеженого розвитку навичок:** Для користувачів, які прагнуть стати професіоналами в програмуванні, може знадобитися перехід до інших мов програмування, оскільки Scratch має обмежені можливості в цьому відношенні.

4. **Не підходить для всіх завдань:** Scratch відмінний для деяких проектів, але не завжди підходить для складних додатків чи завдань, де необхідні високі обчислювальні можливості.

Програма Scratch є потужним інструментом для введення школярів у світ програмування. Візуальний підхід, безкоштовність та доступність, а також активна громадська спільнота роблять її привабливою для широкого кола користувачів. Візуальне програмування полегшує створення програм та сприяє розвитку критичного та логічного мислення. Проте Scratch має свої обмеження. Обмежена складність проектів та відсутність розширених програмувальних мов можуть виявитися недостатніми для вивчення сучасних мов програмування. Для тих, хто прагне до професійного програмування, може знадобитися перехід до інших мов.

Усе ж, загальний вплив Scratch є позитивним, сприяючи навчанню програмування та розвитку творчих навичок. Важливо розуміти, що програма має свою місію в навчанні та стимулюванні інтересу до програмування, а вибір подальших кроків може залежати від конкретних цілей користувача.

РОЗДІЛ 3: SPIKE PRIME ДЛЯ РОБОТОТЕХНІКИ

3.1. Комплект Spike Prime та його застосування

Сучасний освітній процес вимагає активного використання новітніх технологій та методів навчання для стимулювання інтересу учнів до науки та інженерії. Однією з таких інноваційних педагогічних платформ є навчальний комплект LEGO Spike Prime, який розроблено спеціально для використання в освітніх закладах для розвитку навичок STEM (наука, технологія, інженерія та математика) серед учнів.

LEGO Spike Prime – це навчальний комплект, розроблений компанією LEGO Education з метою впровадження цікавого та інтерактивного підходу до вивчення STEM-дисциплін у середній та старшій школі. Цей комплект містить різноманітні деталі, мотори, сенсори та мікроконтролер, які можна об'єднати для створення різноманітних роботів і системи (LEGO Education. Spike Prime [Електронний ресурс]. 2023).

Spike Prime надає учням можливість розробляти, збирати та програмувати різні роботи, що сприяє розвитку їх інженерних навичок. Використання моторів і сенсорів дозволяє створити різноманітні механічні та електронні системи, вивчаючи при цьому закони фізики та математики. Spike Prime використовує мову програмування Scratch, яка є візуальною та легкою для розуміння навіть для початківців. Учні можуть створювати програми для керування роботами та вивчати основи інформатики. Легко змінюючи деталі та програми роботів, учні розвивають креативність та навички проблемного мислення. Вони вчаться шукати рішення для реальних завдань та задач.

Spike Prime є інструментом для підвищення якості STEM-освіти в сучасному світі. Використання цього навчального комплекту сприяє розвитку практичних навичок учнів та підготовці нового покоління фахівців у галузі науки та технології.

Навчальний комплект LEGO Spike Prime є важливим інструментом для розвитку навичок STEM серед учнів. Його можливості у навчанні включають розвиток інженерних навичок, програмування, креативності та проблемного мислення. Використання Spike Prime сприяє підготовці учнів до викликів сучасного світу і забезпечує їм необхідні навички для майбутнього успіху в STEM-галузях.

Spike Prime – це практичний навчальний набір STEAM, призначений для класу, який складається з 528 частин (включно з 11 новими елементами, двигунами та концентратором) і мовою кодування з функцією перетягування.

Spike Prime Hub має новий процесор, який підключається до різних датчиків, зокрема оптичних, сенсорних, шестиосьового гіроскопа з триосьовим акселерометром і триосьового гіроскопа, а також USB-порти розширення та Bluetooth.

Мозок і датчики програмуються за допомогою програми, доступної для iOS, Android і настільних операційних систем. Подібно до Lego Boost, орієнтованого на дітей молодшого віку, Spike використовує графічний метод програмування. Але цей набагато складніший і заснований на Scratch (Spike Prime in programming [Електронний ресурс]. 2023).

3.2. Програмування Spike Prime для створення роботів

Програмування Spike Prime для створення роботів може бути виконано за допомогою середовища програмування Spike Prime, яке надає зручність та можливості для створення різних функцій і завдань для ваших роботів.

За допомогою Lego Education Spike Prime школяр може вивчати робототехніку по шляху наростання складності: робота з базовими моделями, вивчення авторських моделей роботів для розширення знань і знайомства з ідеями різного використання роботів, створення власної розробки. Таким чином, робототехнічні конструктори Lego можуть використовуватися на етапі знайомства школярів з робототехнікою, але з акцентом на те, що це дійсно лише

«перші кроки в робототехніку». Ось загальний опис процесу програмування Spike Prime:

- **Підключення Хабу:** потрібно почати з підключення хаба Spike Prime до вашого комп'ютера або мобільного пристрою за допомогою бездротового зв'язку.

- **Запуск середовища:** відкрийте програмне середовище для програмування Spike Prime на вашому пристрої. Зазвичай це є спеціальним програмним засобом, який може бути завантажений з веб-сайту виробника LEGO Education.

- **Створення програми:** виберіть тип програми, яку ви хочете створити, та розпочніть програмування. Ви можете використовувати блоки програмування для створення послідовностей команд для робота. Додавайте блоки для переміщення, поворотів, взаємодії зі сенсорами тощо.

- **Налаштування сенсорів та двигунів:** під час програмування налаштовуйте сенсори та двигуни робота відповідно до завдань, які ви плануєте виконати. Вказуйте, як робот має реагувати на навколишнє середовище та команди програми.

- **Тестування і відладка:** після створення програми перевірте її на роботі. Відстежуйте, як робот виконує програму та виправляйте помилки, якщо вони з'являються.

- **Завдання і виконання:** Після успішного програмування робота, запускайте його виконання завдань. Дивіться, як робот реагує на програму та виконує поставлені завдання.

Це загальний процес програмування роботів у системі Spike Prime. Залежно від ваших конкретних проектів і завдань, ви можете створювати різноманітні програми для керування роботами та їхньої взаємодії з навколишнім середовищем. Хараджян Н.А., Пихтіна І.О. (2018) описали умови задач, наприклад:

1. Перемістити робота на відстань 1 м від початкового положення.

2. Робот стоїть на чорній лінії кола. Треба об'їхати повністю коло, визначивши наприкінці його діаметр, радіус та довжину.
3. Задається відстань, яку робот має проїхати і час, протягом якого він має це зробити. Після запуску програми робот повинен прибути в кінцеву точку маршруту у точно заданий час.

3.3. Як скласти робота засобами роботехнічного конструктора Spike Prime

Spike Prime – є сучасним конструктором. Набір Lego Spike Prime (рис. 3.1) також представляє деякі нові цеглинки Technic разом із набором, як-от каркас 3x3, який використовується для полегшення будівництва, базову плиту, яку можна використовувати для створення прототипів студентських конструкцій, колеса, які легко кріпити до двигунів, і навіть дротяні затискачі. Коробка Spike Prime містить три двигуни, три датчики та Smart Hub, центральну цеглину, яка містить шість портів, динамік, світлову матрицю 5x5, 6-осьовий гіроскоп, Bluetooth і акумуляторну батарею (Resnick, 2003, 60-67).



Рис. 3.1. Базовий набір Lego Spike Prime

У табл. 3.1 буде представлено деталі яких не має у наборі Lego Education WeDo 2.0. Комплектацію набору Lego Spike Prime представлено в таблиці 3.1.

Таблиця 3.1.

Склад конструктора Lego Spike Prime

	<p>Hub у Spike Prime відіграє ключову роль у функціонуванні робота та надає можливість програмувати його та керувати рухом та взаємодією з навколишнім середовищем.</p>
	<p>Датчик кольору. Може визначати до 8 різних кольорів (зелений, червоний, синій, білий, жовтий, чорний, фіолетовий, блакитний) та має режим не реагування на колір. Також має режим вимірювання відбитого світла.</p>
	<p>Датчик відстані – це сенсор, який вимірює відстань між роботом та об'єктами за допомогою ультразвукових сигналів.</p>
	<p>Датчик сили – це сенсор, який вимірює силу, яку на нього застосовують. Він допомагає роботу реагувати на зовнішні сили та виконувати відповідні дії в залежності від сили, яку він вимірює.</p>
	<p>Великий двигун – це потужний електромеханічний пристрій, який використовується для надання руху роботу, має швидкість 175 об/хв.</p>
	<p>Середній двигун – це середньої потужності електромеханічний пристрій, який використовується для надання руху роботу, має швидкість 135 об/хв.</p>

3.4. Приклади проектів з використанням Spike Prime

Познайомившись із набором та програмою Spike Prime давайте розглянемо приклади моделей та їхній функціонал. Droid bot – це модульний базовий робот, побудований переважно з одним базовим набором Spike Prime і ще одним великим двигуном. Щоб побудувати деякі модулі, вам знадобиться додатковий датчик кольору, кульове колесо і рамку з розширення. Насадки легко одягаються і знімаються. Зразок модуля кріплення надається.

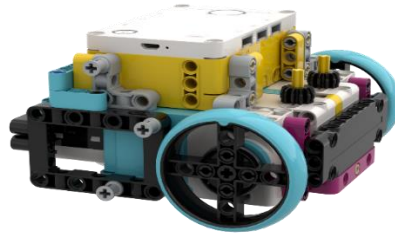


Рис. 3.2. Модель: «Droid bot»

Наступний представник «Локомотив» (рис. 3.3). Модель складніша у конструюванні, але представлена більша кількість різних механізмів. Наприклад: в локомотиві реалістично і асинхронно можуть працювати поршень і клапан.

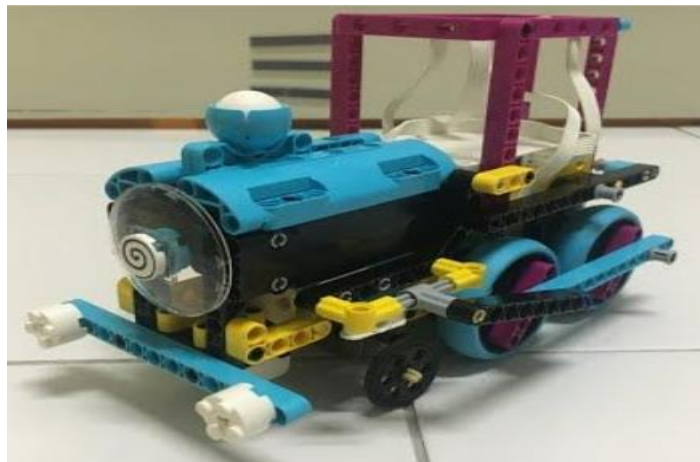


Рис. 3.3. Модель: «Локомотив»

«Робот-танцівник» (рис. 3.4) – може бути сконструйований для того, щоб побачити ритмічні рухи і програмування циклів. робот може починати танцювати з певного стартового руху або позиції. Як ускладнення робота – використовують сенсор звуку для визначення темпу музики і змінюють відповідно танцювальні рухи, додають випадкові елементи до програми, щоб робот виконував різні танцювальні рухи при кожному виконанні.



Рис. 3.4. Модель «робот-танцівник»

3.2. Переваги та недоліки програми Spike Prime

Програма Spike Prime має свої переваги та недоліки, що варто враховувати при її використанні. Однією з переваг є інтеграція з апаратним забезпеченням LEGO Education, що дозволяє вивчати робототехніку в закладах загальної середньої освіти. Spike Prime надає широкий функціонал для створення різноманітних роботів та вчить основам програмування, сприяючи розвитку творчих навичок учнів. Також до позитивних аспектів можна віднести відносно просту користувача та дружній інтерфейс, що дозволяє швидко освоїти основи робототехніки навіть початківцям. Розглянемо інші переваги програми детальніше:

1. **Легкість використання:** Spike Prime розроблений для освіти і легкий у використанні, що робить його перспективним для навчання робототехніки та програмування.

2. **Всебічний набір сенсорів:** Комплект постачається з різними сенсорами, такими як сенсори відстані, кольору та сенсори касання, що дозволяє створювати різноманітні роботи.

3. **Інтердисциплінарність:** Spike Prime дозволяє поєднувати робототехніку з STEM-предметами, що сприяє більш глибокому розумінню науки та технології.

4. **Гнучка система програмування:** Spike Prime підтримує не тільки графічне програмування, але й Python, що дозволяє розвивати програмувальні навички на різних рівнях складності.

5. **Широкі навчальні ресурси:** Існує багато навчальних матеріалів і програм для використання з Spike Prime, які варто використати в навчанні.

Програма має дуже багато спільного зі Scratch. Перше і основне це візуальне програмування. Учні які до цього займалися в Scratch зможуть дуже легко освоїти інтерфейс програми. Програма також складається із блоків які потрібно переносити на сцену і зв'язувати їх між собою. Завдяки більшій кількості датчиків Spike Prime має більший функціонал. Тому можливості, які обмежував Scratch тут зникають і учні мають змогу придумувати щось більше.

Результати порівняння двох програм записані в табл. 3.2.

Таблиця 3.2.

Порівняльна таблиця Scratch та Spike Prime

Scratch	Spike Prime
Призначений для всіх вікових груп, у програмуванні та створенні креативних проектів.	Спеціально розроблений для освіти, з орієнтацією на навчання робототехніки та STEM-предметів у школах.
Використовує блокову мову програмування, що робить його дуже доступним у програмуванні.	Підтримує графічне програмування, а також мову програмування Python, що дає більше можливостей для розвитку навичок.
Використовується для створення ігор, анімацій, інтерактивних історій та проектів в області мистецтва та креативності.	Використовується для навчання робототехніки та STEM-предметів, а також для створення фізичних роботів, які взаємодіють із середовищем.
Має активну громадську спільноту користувачів і безліч навчальних матеріалів у мережі.	Має обмежену спільноту користувачів, але вона збільшується чи не щодня, а також навчальні ресурси для освіти.
Використовується для розвитку обчислювального мислення та креативності в навчанні.	Спеціально розроблений для освіти, з фокусом на STEM-предметах та робототехніці в школах.

3.6. Приклади програм у Spike Prime

Дивлячись на те, що програми Scratch Spike prime подібні, то будемо розглядати програми які можливо створити лише у Spike Prime.

Особливість даної програми полягає в тому, що перед початком роботи потрібно вказати які пристрої ми будемо використовувати. Наприклад, програма рис. 3.4. буде працювати наступним чином: спершу обираємо мотори А та В, далі встановлюємо швидкість на 50% (визначає на скільки мотор буде швидко обертатися) і в кінці вказуємо шлях для моторів. Оскільки в хаб вбудований гіроскоп тому шлях вказується за допомогою стрілок, а вже сам робот розуміє в якому напрямку повинен рухатися. Беручи у відповідність програму Scratch, мали можливість лише бачити, що за рух моторів відповідає те, що можливо вибрати варіант напрямку руху за годинниковою стрілкою або проти. У цій програмі можливо побачити, що маємо більший функціонал.

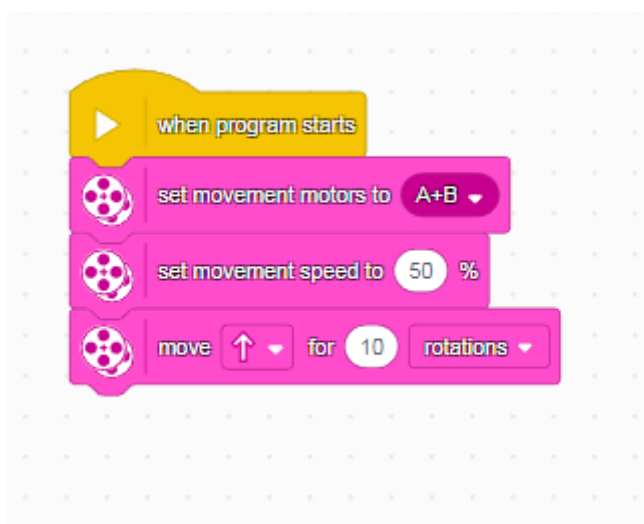


Рис. 3.4. Приклад програми руху у Spike Prime

Вдосконалимо програму наступним чином. Використаємо можливість показувати зображення на екрані хабу та додаємо відлік перед початком старту роботи моторів (рис. 3.5).

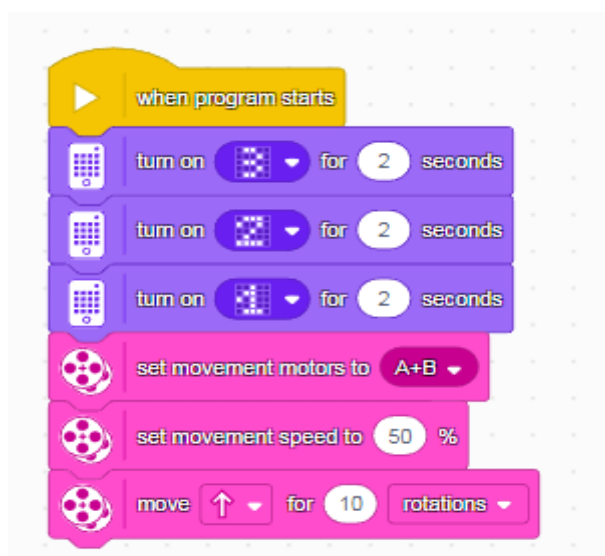


Рис. 3.5. Відлік на екрані хабу перед початком руху

Додаємо датчик сили до хабу. Використаємо його для того, щоб при його натисканні робот почав рухатися, а якщо він не буде задіяний, то робот буде зупинений (рис. 3.6).

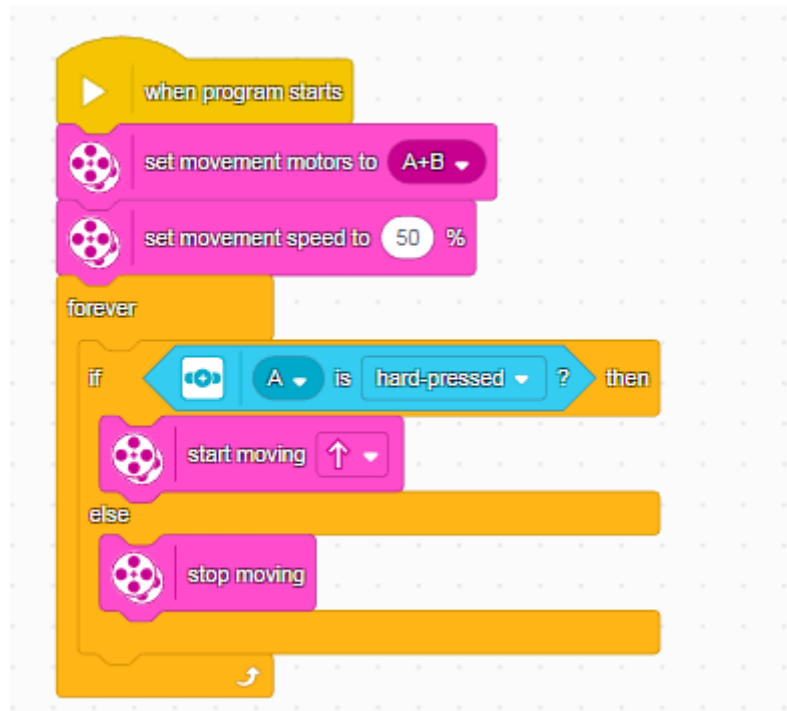


Рис. 3.6. Рух робота через датчик сили

Також, перед початком роботи, можемо побачити, що вказуємо мотори які запускаємо. Не обравши мотори програма працювати не буде.

Використовуючи конструктори LEGO та електронний блок, Spike Prime дозволяє учням створювати робототехнічні проекти. Він використовує власну візуальну середу програмування, де користувачі можуть з'єднувати блоки коду для керування роботами та виконання різних завдань. У відмінності від Scratch, який акцентується на введенні у програмування через групу блоків, Spike Prime включає аспекти фізичного конструювання роботів, що дозволяє об'єднати програмування та інженерію.

ВИСНОВКИ

Використання програм Scratch та Spike Prime, виявляється ефективним засобом розвитку мислення учнів. Завдяки цим інструментам, навчання стає захопливим та цікавим процесом, що сприяє активізації когнітивних функцій учнів.

Програма Scratch дозволяє учням створювати власні комп'ютерні програми, використовуючи блоки та графічний інтерфейс. Цей підхід сприяє розвитку логічного та алгоритмічного мислення. Учні вчаться розуміти послідовність дій, формулювати завдання, та ефективно вирішувати проблеми, сприяючи розвитку креативності та критичного мислення.

Система Spike Prime, використовуючи роботів та сенсори, дозволяє учням навчатися в реальному середовищі. Це стимулює їхню увагу та інтерес до навчання, а також сприяє розвитку просторового та інженерного мислення. Учні навчаються працювати з технологічними рішеннями, розвиваючи навички програмування, а також вирішення реальних завдань.

Таким чином, використання освітньої робототехніки в навчальному процесі, зокрема за допомогою програм Scratch та Spike Prime, сприяє розвитку різноманітних аспектів мислення учнів. Це не лише розвиває їх технічні навички, а й сприяє формуванню креативності, аналітичного та системного мислення, що є важливими компонентами сучасної освіти.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Riener R. (2016). The Cybathlon promotes the development of assistive technology for people with physical disabilities. *Journal of NeuroEngineering and Rehabilitation*, 49.
2. Laschi C., & Cianchetti M. (2016). Soft robotics: new perspectives for robot bodyware and control. *Frontiers in Bioengineering and Biotechnology*, 3. URL: <https://www.frontiersin.org/articles/> (date of access: 15.10.2023)
3. Levine S., Finn C., Darrell T., & Abbeel P. (2016). End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 1-40.
4. Робототехніка у середній школі [Електронний ресурс]. 2023. URL: <https://inventor.com.ua/> ([дата звернення 25.10.2023](#)).
5. Баранов С. (2020). Про актуальність розробки класифікації засобів навчання освітньої робототехніки. ББК. Математичне моделювання та розробка програмного забезпечення.
6. Гойда В.О. & Павлова Н.С. (2023). Робототехніка в новій українській школі. Матеріали XVI Всеукраїнської науково-практичної конференції «Інформаційні технології в професійній діяльності». URL: https://iktmvi.rshu.edu.ua/files/konf/Zbirnyk_ITvPD_Rivne_1-11-2023.pdf
7. Кривонос О.М. (2023) Проблеми навчання робототехніці як одного з компонентів STEM-освіти.
8. Гойда В.О & Павлова Н.С. (2022) Робототехніка як сучасний засіб навчання. Матеріали педагогіка сучасності: виклики і перспективи цифрової доби.
9. Anderson M., & Anderson S. L. (2010). Machine ethics: Creating an ethical intelligent agent. *AI & Society*, 293-299.
10. Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.
11. Introduction to Scratch Programming Language [Електронний ресурс]. 2023.
12. Пауль Р., & Елдер Л. (2006). Критичне мислення: Природа критичного та творчого мислення,

13. Терно С. (2012) Методика розвитку критичного мислення школярів у процесі навчання історії, Запоріжжя.
14. Рензулі Д. С. (2012). Трикільцеве уявлення про обдарованість: Розвиваюча модель для підтримки творчої продуктивності.
15. Козир М.В. & Грабовська М.О. (2019) Уміння доводити власну думку в процесі розвитку критичного мислення молодших школярів, Педагогічна освіта, с. 8.
16. Конотоп К.О. & Матвієнко Ю.С. (2023) Особливості викладання освітньої робототехніки дітям молодшого шкільного віку. Матеріали XLVI Міжнародної наукової студентської конференції за підсумками науково-дослідних робіт студентів за 2022 рік. С. 715.
17. Кузьменко С. В. (2017). Робототехніка в школі / С.В. Кузьменко, Є.В. Кузьменко, О.І. Хомутовський // Збірник матеріалів V Всеукраїнської науково-практичної конференції молодих учених «Наукова молодь-2017». К.: ІТЗН НАПН України,.
18. Scratch URL: <https://scratch.mit.edu> (date of access: 15.10.2023)
19. Хараджян Н.А., Пихтіна І.О. (2018) Розробка системи задач для розвитку конструкторського мислення учнів середньої та старшої школи на заняттях з робототехніки. Новітні комп'ютерні технології Т. XVI. С.235-238
20. Гойда В.О. & Павлова Н.С. (2021) Робототехніка засобами Scratch. Матеріали XIV Всеукраїнської науково-практичної конференції «Інформаційні технології в професійній діяльності».. С. 96. URL: https://iktmvi.rshu.edu.ua/files/konf/Zbirnyk_ITvPD-2021.pdf (дата звернення: 15.10.2023)
21. Introduction to Scratch Programming Language URL: <https://piembsystech.com/introduction-to-scratch-programming-language> (date of access: 17.10.2023)
22. LEGO Education. Spike Prime. URL: <https://education.lego.com/en-us/products/spike-prime-set/4567> (date of access: 17.10.2023)

23. Brennan K. & Resnick M. (2012). New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 Annual Meeting of the American
24. Що таке Scratch та як його використовувати. URL: <https://buki.com.ua/news/skretch-yak-vykorystovuvaty> (date of access: 17.10.2023)
25. Візуальне програмування – що це? URL: <http://www.itschool.vn.ua/visual-programming-about> (дата звернення: 18.10.2023)
26. LEGO Education celebrates 40 years by launching Spike Prime engineering kit worldwide URL: <https://www.brothers-brick.com/2020/01/14/lego-education-celebrates-40-years-by-launching-spike-prime-engineering-kit-worldwide-news> (date of access: 17.10.2023)
27. Spike Prime in programing URL: <https://www.theverge.com/2019/4/2/18292326/lego-spike-prime-programming-bricks-scratch-technic-robotics> (date of access: 18.10.2023).
28. Resnick, M. (2003). Scratch: Programming for All. Communications of the ACM, 52(11), 60-67.
29. Халперн Д. Ф. (2014). Мислення та знання: Вступ до критичного мислення, Запоріжжя, , 5-18.
30. Гойда В.О. & Павлова Н.С. (2023). Робототехніка, як об'єкт вивчення у закладах освіти. Матеріали II Всеукраїнської наукової конференції «Підготовка педагогів до професійної діяльності в умовах змішаного навчання». С. 98. URL: https://iktmvi.rshu.edu.ua/files/konf/Zbirnyk_PPdoPDvUZN_2023.pdf (дата звернення: 18.10.2023)
31. Кивлюк О.П., (2000) Використання комп'ютера на уроках інформатики в початковій школі. Комп'ютер у школі та сім'ї.. №4, 33 - 40.

ДОДАТКИ

Додаток А

Сертифікати учасника конференцій



Рис. А.1. Сертифікат учасника конференції




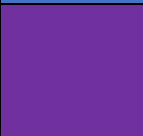

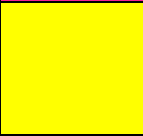






Рис. А.2. Сертифікат учасника конференції



Рис. А.3. Сертифікат учасника конференції

Додаток Б
Таблиця Б.1

Короткий опис блоків у програмі Scratch

Група	Англійська назва	колір	примітка
Рух	Motion		Керує рухом спрайту .
Зовнішній вигляд	Looks		Керує зовнішністю спрайту.
Звук	Sound		Керує звуком спрайту.
Події	Events		Наявності подій, відправка сигналів до всіх спрайтів.
Управління	Control		Керуючі конструкції, заголовки обробників подій.
Сенсори	Sensing		Опитування пристроїв введення, таймер та ім'я учасника.
Оператори	Operators		Відображає Арифметико-логічні операції.
Змінні	Variables		Розділ для керування змінними та списками.
Інші блоки	My Blocks		Перетворює комбінацію блоків на один блок, що спрощує код,
Додати розширення	Extension		Розширює можливості Scratch: музика, Makey Makey Lemo Mindstorms EV3 та Lego Education WeDo 2.0

РОБОТОТЕХНІКА, ЯК ОБ'ЄКТ ВИВЧЕННЯ У ЗАКЛАДАХ ОСВІТИ

РОБОТОТЕХНІКА, ЯК ОБ'ЄКТ ВИВЧЕННЯ У ЗАКЛАДАХ ОСВІТИ

Гойда В. О.

здобувач другого (магістерського) рівня вищої освіти
Рівненський державний гуманітарний університет

Павлова Н. С.

кандидат педагогічних наук, доцент, професор кафедри інформаційно-комунікаційних технологій та методики викладання інформатики
Рівненський державний гуманітарний університет

Анотація. Охарактеризувати робототехніку, як об'єкт вивчення у закладах освіти; описати напрямки вивчення робототехніки у закладах загальної і позашкільної освіти та конкретизовано тематику занять для майбутніх учителів інформатики.

Ключові слова: робот, робототехніка, Lego, майбутній вчитель інформатики.

Vyacheslav Hoida, Natalia Pavlova. Robotics as an object of study in educational institutions

Abstract. Describe robotics as an object of study in educational institutions; to describe the directions of studying robotics in institutions of general and extracurricular education and the specific subject of classes for future computer science teachers.

Key words: robot, robotics, Lego, future computer science teacher.

Інформаційне суспільство сприяє розвитку такої прикладної галузі як робототехніка. Відповідно з'являється потреба: вивчати **робототехнічні** системи, формувати навички їх конструювання, програмування та використання; «Підготовка педагогів до професійної діяльності в умовах мішаного навчання» 99 розширювати уявлення про науково-технічний прогрес та сучасні професії у галузі робототехніки; розвивати інженерне мислення.

Аналіз стану навчання робототехніки в закладах освіти показав, що її вивчення здійснюється: окремими темами на **уроках** інформатики або технологій у закладах загальної середньої освіти; за спеціально розробленими програмами гурткових занять у позашкільних закладах за інтересами дітей з урахуванням вікової категорії; окремою спеціальністю чи дисциплінами «Освітня робототехніка», «**Робототехнічні системи**», «**Step-освіта**», «Штучний інтелект» та ін. у закладах вищої освіти. Як бачимо, робототехніка є універсальним інструментом для освіти, вона доцільно вписується у вищу освіту, в позашкільне навчання, у навчальні предмети шкільної програми. Крім того, робототехніка може бути використана для інтеграції з іншими дисциплінами навчального плану, створюючи міжпредметні зв'язки та сприяючи глибокому розумінню матеріалу. Наприклад, здобувачі можуть створювати роботів, які вивчають екосистеми в рамках біологічних наук або роботів, що моделюють фізичні закони.

Спостерігаємо масове вивчення робототехніки у гуртках **інформаційнокомп'ютерного** профілю, вихованцями яких є діти віком від 10 до 15 років. Складність занять зумовлена використанням технічного і програмного забезпечення. На ринку робототехніки є різні платформи, спеціально розроблені для впровадження в освітній процес. Так, у закладах загальної і позашкільної освіти використовують **робототехнічні** набори на основі **Arduino**, конструктори і платформи від компанії **Lego**. Навчання спрямоване на складання роботів, кожен з яких ототожнюється із механічним пристроєм, який запрограмований для виконання інструкцій.

Вивчення основ робототехніки у заклади загальної і позашкільної освіти змушує формувати відповідні знання у студентів як майбутніх учителів інформатики. Щоб ефективно викладати робототехніку, вчителі повинні 100 Матеріали II Всеукраїнської науково-практичної конференції отримати відповідну професійну підготовку – сформувані знання щодо необхідного обладнання, матеріалів та програмного забезпечення [1]. З цієї метою у Рівненському державному гуманітарному університеті бакалаври вивчають

дисципліни «Освітня робототехніка» (обов'язковий курс), «Вибрані питання STEM-освіти» (вибірковий курс). Дисципліна «Освітня робототехніка» відображена 3-ма кредитами, тобто 90 годин (з них 18 – лекційних, 18 – лабораторних і 54 – самостійна робота студентів). Вивчення робототехніки сприяє розвитку творчості та інноваційного мислення. Студенти створюють власні проекти, вдосконалюють ідеї та шукають нові способи використання робототехніки в різних сферах життя [2]. Тематика лекційних і лабораторних занять перегукуються:

- **stem-освіта** та робототехніка;
 - системи автоматизованого проектування (використання комп'ютерних програм для створення, моделювання, аналізу та оптимізації дизайну продукту; формування навичок використання спеціалізованих програм для проектування; вивчення 2D і 3D моделювання);
 - механізми робота (будова робота і принципи його теоретичної роботи);
 - фізичні величини в **робототехніці** (вивчення фізичних величин, які впливають на поведінку роботів, **наприклад**: рух, енергія, обертання);
 - середовище програмування, програмна реалізація робота (це програмна реалізація функціональності роботів. Це означає створення програмного коду, який визначає, як роботи повинні виконувати різні завдання та взаємодіяти з навколишнім середовищем);
 - можливості робототехніки, **робототехнічні** системи в різних галузях.
- Загалом, робототехніка є сучасною, цікавою та практичною галуззю вивчення у закладах освіти, яка сприяє розвитку **компетентностей**, стимулює творчий та інноваційний підхід до вирішення проблем. «Підготовка педагогів до професійної діяльності в умовах мішаного навчання»

Список використаних джерел

1. Робототехніка [Електронний ресурс]. 2023. URL: <https://uk.economypedia.com/> (дата звернення 12.05.2023);
2. Робототехніка у середній школі [Електронний ресурс]. 2023. URL: <https://inventor.com.ua/> (дата звернення 12.05.2023).

References

1. Robotics [Electronic resource]. – 2023. – Retrieved from: <https://uk.economypedia.com/> [in Ukrainian];
2. Robotics in secondary school [Electronic resource]. – 2023. – Retrieved from: <https://inventor.com.ua/> [in Ukrainian].

Рис. В.1 Тези автора дослідження

РОБОТОТЕХНІКА В НОВІЙ УКРАЇНСЬКІЙ ШКОЛІ

РОБОТОТЕХНІКА В НОВІЙ УКРАЇНСЬКІЙ ШКОЛІ

Гойда В. О.

здобувач другого (магістерського) рівня вищої освіти
Рівненський державний гуманітарний університет

Павлова Н. С.

кандидат педагогічних наук, доцент, професор кафедри інформаційно-комунікаційних технологій та методики викладання інформатики
Рівненський державний гуманітарний університет

Анотація. Досліджено аспекти та функції вивчення робототехніки в новій українській школі. Візуалізовано ідеї вивчення робототехніки, наведено зразки роботів.

Ключові слова: робототехніка, робот, урок інформатики.

Novda Vyacheslav, Pavlova Nataliya. Robotics in a new Ukrainian school

Abstract. The aspects and functions of studying robotics in a new Ukrainian school are investigated. The ideas of studying robotics are visualized, and samples of robots are presented.

Key words: robotics, robot, computer science lesson.

Сучасний освітній процес у закладах загальної середньої освіти характеризується впровадженням робототехніки через, наприклад, уроки інформатики, фізики, технологій, гурткові і факультативні заняття. Форми організації навчання визначаються різними чинниками: від напрямку вивчення, вікової категорії учнів до робототехнічних конструкторів та інших компонентів матеріально-технічної бази закладу освіти. Метою введення робототехніки в навчальні плани з інформатики і технологій є формування в учнів обізнаності щодо моделювання, конструювання, програмування роботизованих систем, а також встановлення міждисциплінарних зв'язків інформатики з іншими галузями.

Існує декілька напрямків вивчення робототехніки: *free model* (творче конструювання роботів без поглиблення у механізми та алгоритми); *half models* (вивчення вузлів, датчиків, сенсорів, процесорів та інших елементів в одній конструкції, наприклад, в інтерактивному роботі); *one model* (проєктування, складання, програмування робота та його різностороннє вдосконалення) (рис.1) [1].

Створення різних конфігурацій роботів (рис. 2) та їх програмування не лише підвищують інтерес до технічної галузі знань, а і сприяють розвитку *soft skills* (генерування творчих ідей, застосування прийомів критичного мислення, виконання групових проєктів), поглиблюють знання про алгоритми та їх програмування у певних середовищах.

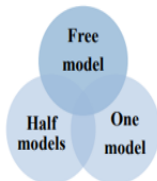


Рис. 1. Ідеї вивчення робототехніки на уроках інформатики



Рис. 2 Зразки роботів засобами конструктора Lego

Освітня робототехніка розвивається і стає дедалі популярнішою, тому вчителі повинні спиратися на готовність та здатність поглиблювати й оновлювати власні знання й уміння в цій галузі.

Список використаних джерел

1. Робототехніка у середній школі. 2023. URL: <https://inventor.com.ua/> (дата звернення 25.10.2023).

Рис. В.2. Тези автора дослідження

РОБОТОТЕХНІКА ЗАСОБАМИ SCRATCH

РОБОТОТЕХНІКА ЗАСОБАМИ SCRATCH

Гойда В. О.

студент групи І-41 факультету математики та інформатики

Павлова Н.С.

кандидат педагогічних наук, доцент

Рівненський державний гуманітарний університет

Анотація. Описано особливості роботи програми *Scratch*, обґрунтовано вивчення мови програмування школярами. Акцентовано увагу на інтеграції *Scratch* із робототехнікою.

Ключові слова: *Scratch*, робототехніка, програмування.

Hoyna Viacheslav, Pavlova Natalia. Robotics by Scratch

Abstract. The peculiarities of *Scratch* program work are described, the study of programming language by schoolchildren is substantiated. Emphasis is placed on the integration of *Scratch* with robotics

Key words: *Scratch*, robotics, programming

Scratch є інтерпретовано динамічно візуальною мовою програмування. Сьогодні зростає популярність цієї мови програмування завдяки широкому впровадженню в загальноосвітні заклади та позашкільну освіту з метою формування в учнів уявленн про програмування, вироблення відповідних навичок та розвитку алгоритмічного мислення. Масштабність такого застосування мови зумовлена тим, що ця мова програмування є вільно поширеним програмним продуктом, таким чином, будь-який користувач може завантажити програму з Інтернету і приступити до вивчення особливостей роботи у середовищі програмування.

Розробники *Scratch* взяли до уваги зручні аспекти обчислювальної техніки і дизайну інтерфейсів різних програмних засобів задля того, щоб зробити процес програмування більш привабливим і доступним для дітей шкільного віку і тих, хто хоче навчитися програмуванню. Наприклад, учні початкової школи, працюючи з *Scratch* можуть усвідомити зміст таких понять як «паралельність» і «синхронізація».

Використовуючи засоби *Scratch* учні створюють власні анімовані та інтерактивні історії, ігри та інші «продукти» технологічного простору. Школярі, як користувачі *Scratch* відзначають наступну перевагу середовища – можливість переглянути проекти, обмінятися ідеями і розробками у міжнародній спільноті, яка сформована в мережі Інтернет. Таке спільне інформаційне середовище є одним із найбільших досягнень розробників мови *Scratch*.

Іншими основними особливостями *Scratch* є:

1) блочне програмування. Для створення програм потрібно поєднати графічні блоки.

Блоки зроблені так, що їх можна зібрати лише в стеках в синтаксично вірних конструкціях, уникаючи таким чином можливих помилок на етапі розробки програми. Різні типи даних мають різні форми, підкреслюючи несумісність окремих типів. Наявність такої характеристики, як динамічність дозволяє розробнику змінювати код навіть під час виконання програми, таким чином з'являється більше можливостей для експериментування з новими ідеями;

2) маніпуляції з даними. *Scratch* дозволяє користувачу створювати програми, які поєднують графіку, анімацію, музику та звуки. *Scratch* розширює можливості управління візуальними даними, використання яких в сьогодиншій інформаційній культурі є популярним.

Учням, які усвідомили функціонування програми та засвоїли початкові навички програмування на певному етапі роботи пропонується програмувати роботів. Для виконання такої діяльності середовище *Scratch* має безліч додаткових функцій, зокрема додавання інших частин, однією з яких є *SmartHub*. *SmartHub* *WeDo* 2.0 – це процесор, який є об'єктом програмування, керуючи датчиками і двигунами. Він виконаний в стилі «розумної цеглини»

EV3, але істотно меншого розміру. Блок не дозволяє програмувати на ньому самому, має одну кнопку, яка відповідає за вклучення і запуск пошуку блоку комп'ютером чи планшетом згідно протоколу *Bluetooth* 4.0. Програмування здійснюється за допомогою комп'ютера або планшета після підключення.

На перших етапах роботи діти навчаються конструювати моделі, які можна запустити за допомогою комп'ютерної програми. Спочатку вони копіюють програму, написану викладачем, але поступово навчаються писати власні. А вже на початкових етапах це не складно – для таких моделей використовують візуальне програмування, де кожна дія запускається за допомогою блоку-картинки. Блок із зображенням мотора та поворотною стрілкою направо означає обертання мотора за годинниковою стрілкою. Для старших програмування стає складніше, а сама робототехніка перестає бути модульною (з готових конструкторів). Поступово учні починають знайомитися з електронікою і дізнаються, як влаштовані різні електронні компоненти і створюють робота власноруч.

За допомогою того, що діти відвідують заняття з робототехніки, вони проєктують власні історії, ігри, анімації у середовищі *Scratch* і таким чином навчаються комп'ютерної грамотності здобувати навички програмування, зокрема:

- зрозуміють термінологію мови програмування, структури блокових мов та принципу
- їх роботи;
- навчатися визначити переваги та недоліки використання різних механізмів та електронних компонентів для моделі робота;
- створювати комп'ютерні ігри та власні програми для роботи;
- розвинути технічне мислення, уміння аналізувати та працювати у команді.

Робототехніка формує обізнаність про автоматизовані системи, які діють за задальгідь створеними програмами. Роботи непомітно оточують нас, зокрема, на виробництві та у побуті. У майбутньому з'являться безліч професій, пов'язаних із робототехнікою. Це будуть інженери, дизайнери, тестувальники, майстри з ремонту машин, програмісти роботів, що вийшли з ладу і так далі. Такими чинниками зумовлена актуальність вивчення робототехніки.

Список використаної літератури

1. *Scratch*: блок вчителя інформатики *Сторожинської С. М.* URL: http://teacheinf.blogspot.com/p/blog-page_85.html (дата звернення 24.10.2021).

References

1. *Scratch*: block of computer science teacher *Storozhynska SM.* URL: http://teacheinf.blogspot.com/p/blog-page_85.html (access date 24.10.2021).

Рис. В.3. Тези автора дослідження

QR-посилання на магістерську роботу