

Працюючи над Скретч-проектами, початківці мають можливість вивчати концепції програмування, основні алгоритмічні структури та отримувати навички програмування додатків.

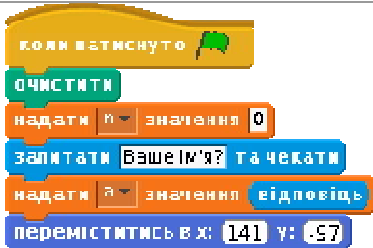

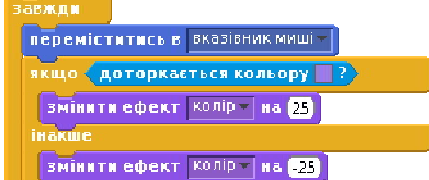
Проектування програм та основи програмування

- генерація ідей - від початкової концепції до комплексного проекту;
- опис структури та поведінки системи, що проектується, визначення очікуваних результатів;
- алгоритмічне мислення;
- навички та досвід створення інтерфейсу користувача;
- налагодження та тестування програм, дослідження результатів.


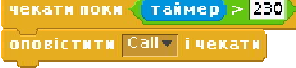
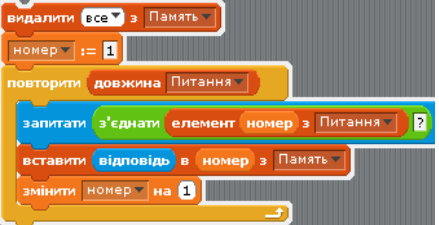
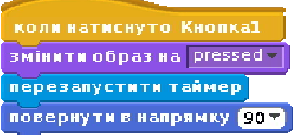
Фундаментальні ідеї щодо комп'ютерів та програмування

- програма однозначно задає комп'ютеру послідовність дій, які необхідно виконати;
- розробка алгоритму розв'язування задачі передує створенню програми;
- чіткий і ретельно продуманий алгоритм – запорука ефективної програми;
- алгоритм не залежать від конкретної мови програмування, а залежать від змісту завдання та методів його розв'язання .

Основні поняття програмування

Поняття	Пояснення	Приклад
Лінійний алгоритм (слідування)	Деякі Скретч-проекти реалізуються з використанням лінійних алгоритмів. Характерними ознаками таких алгоритмів є те, що всі його команди виконуються в заданій послідовності, виконання кожної команди є обов'язковим та однократним.	
Цикли (повторення)	Блоки завжди і повторити можуть використовуватися для повторення деякої послідовності команд.	
Розгалуження (прийняття рішень)	Блоки якщо і якщо-інакше дозволяють реалізовувати алгоритм з розгалуженням.	

Поняття	Пояснення	Приклад
Логічні оператори	<i>і, або, не</i> - логічні оператори, які використовуються в Скретч.	
Випадкові числа	<i>Вибрати випадкове</i> число дозволяє генерувати випадкові цілі числа із заданого інтервалу.	
Змінні	<p>Блоки групи Змінні дозволяють створювати нові змінні та використовувати їх в програмах.</p> <p>Значеннями змінних можуть бути числа або рядки (в неявному вигляді використовуються логічні значення). Підтримуються глобальні та локальні змінні.</p>	
Списки (одновимірні масиви)	<p>Блоки групи Змінні дозволяють створювати та опрацьовувати списки. Така структура даних може розглядатися як "динамічний масив". Значеннями елементів списку можуть бути числа або рядки. На елементи можна посилатися за допомогою імені масиву та індексів.</p> <p>Елементи списку можна імпортувати з текстового файлу формату .txt та експортувати в текстовий файл.</p>	
Опрацювання подій	<p>Виконавці можуть реагувати на певні події: клацання кнопкою мишки, натиснення клавіші на клавіатурі, доторкання до іншого виконавця тощо.</p> <p>В скриптах необхідно передбачити реакцію (опрацювання подій) виконавців на ті чи інші події, викликані користувачем або іншим скриптом.</p> <p>Основним інструментом опрацювання подій є блоки заголовків стеків.</p>	

Поняття	Пояснення	Приклад
Потоки (паралельні процеси)	В проектах виконавці можуть діяти як послідовно, тобто один за одним, так і одночасно, тобто паралельно. Наприклад, при виборі кнопки  одночасно (паралельно) виконуються всі стеки всіх виконавців, першим блоком яких є  . Запускаючи одночасно на виконання n стеків, ми створюємо n незалежних потоків, які виконуються паралельно.	 
Синхронізація (взаємодія виконавців)	Блоки оповістити та коли одержую дозволяють синхронізувати та скоординувати дії кількох виконавців.	Наприклад, виконавець 1 відпра- повідлення Call:  Виконавець 2, отримавши це повідомлення, починає виконувати власну послідовність команд, наприклад: 
Введення даних з використанням клавіатури	Блок Запитати та чекати використовується для отримання в діалоговому режимі даних від користувача (вводяться з клавіатури)	
Взаємодія в режимі реального часу	Блоки групи Датчики та змінні можуть використовуватися для організації динамічної взаємодії в режимі реального часу.	
Проектування інтерфейсу користувача	В Скретч елементами інтерфейсу користувача можуть бути різні фони, активні і пасивні виконавці, клавіші, монітори змінних і т.і.	

Поняття програмування, що не реалізовані в Скретч

процедури і функції; рекурсія; наслідування; визначення класів об'єктів; опрацювання виняткових ситуацій; передача параметрів і отримання результатів; робота з файлами.