Урок 28

Тема: складання та виконання алгоритмів з повторенням і розгалуженням у визначеному навчальному середовищі виконання алгоритмів Scratch.

Мета: навчитися складати та виконувати у проектах Scratch алгоритмічні структури розгалуження та повторення, вдосконалити вміння виділяти множини, визначати зв'язки між об'єктами, представляти їх у нових взаємозв'язках, розчленовувати складні об'єкти на більш прості; формувати обізнаність про можливості ПК як інструменту навчально-пізнавальної діяльності. Після виконання роботи учень

знає:

поняття «вказівка», «виконавець»;
вимоги до вказівок;
поняття алгоритму;
властивості алгоритмів;
вказівки головного меню програми Scratch;
режим перегляду проекту;

має уявлення про:

типи алгоритмів;
поняття розгалуження;
поняття алгоритму з розгалуженням;
поняття циклу;
поняття алгоритму з циклом;

вміє:

завантажувати програму Scratch;
переглядати наявні проекти;
викликати довідку.

Обладнання: ПК з встановленою ОС і середовищем Scratch.

Структура уроку

Організаційний момент.
Актуалізація опорних знань.
Інструктаж з ТБ.
Вироблення практичних навичок.
Підбиття підсумків уроку.
Домашнє завдання.

Хід уроку

1. Організаційний момент

Вітання з класом. Перевірка присутності і готовності учнів до уроку. Перевірка виконання домашнього завдання.

2. Актуалізація опорних знань

Описати:
поняття алгоритму;
способи подання алгоритму;
основні блоки умовних операторів категорії *Керувати*;
основні блоки операторів повторення категорії *Керувати*;
призначення категорії *Змінні*.

3. Інструктаж з ТБ

4. Вироблення практичних навичок

Примітка. Після виконання кожного із завдань повідомляти вчителя підняттям руки.

Завдання 1. Створити проект, в якому над нічним містом під музику спрайт *Кажан* (bat1-a) рухається ліворуч і праворуч.

Вказівки до роботи

Запускаємо програму Scratch.

Видаляємо із проекту Спрайт 1.

Додаємо в проект новий Спрайт. Натискаємо кнопку *Новий об'єкт*, у теці Animals та вибираємо Спрайт *bat1-a*.

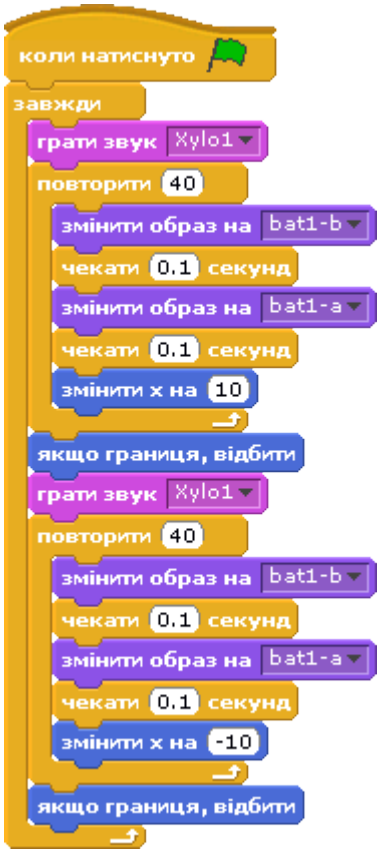
Додаємо для Спрайта *Кажан* новий костюм *bat1-b*.

Переміщуємо Спрайт *Кажан* ліворуч (з цього місця він і розпочне політ) і натискаємо на кнопку *Приймати тільки з ліва на право*.

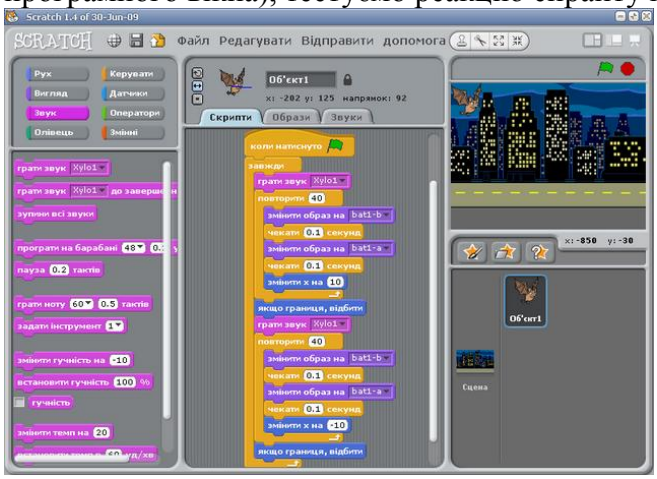
Змінюємо стандартне біле тло на *night-city-with*.

Імпортуємо до бібліотеки звук *Xylo1*.

Створюємо скрипт для нашого об'єкта — послідовність вказівок, що визначають дії та порядок їх виконання об'єктом.



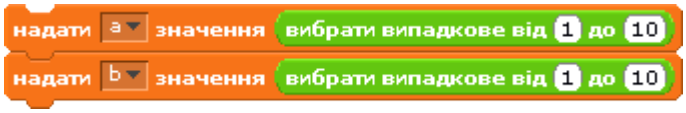
Запускаємо проект на виконання, натиснувши на зелений прапорець у правому верхньому куті програмного вікна), тестуємо реакцію спрайту на натискання клавіші миші.



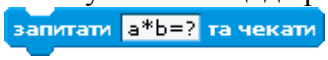
Завдання 2. Створити проект, в якому було б реалізовано роз'язування такої задачі. Пес вирішив з'ясувати, чи знаєте ви таблицю множення. Для цього він кілька разів (наприклад, 5) подасть приклади на множення чисел від 1 до 10 і перевіряти вашу відповідь.

Вказівки до роботи

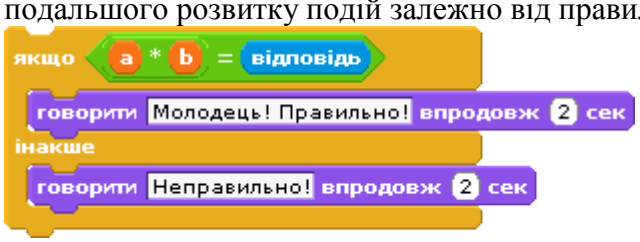
У проекті потрібно використати дві змінні *a* та *b*, які випадковим чином потрібно надавати величини.



Має бути запит щодо результату множення чисел.



Після отримання відповіді потрібно здійснити перевірку відповіді на істинність (правильність). Для цього нам знадобиться конструкція *Якщо-інакше*. Саме тут потрібно прописати варіанти подальшого розвитку подій залежно від правильності відповіді.



Скрипт

Запускаємо проект на виконання, натиснувши на зелений прапорець у правому верхньому куті програмного вікна.

Подумайте!

Що необхідно змінити, аби значення змінних *a* та *b* з'являлись не у віконцях програми, а безпосередньо в запитанні?

Як змінити скрипт, щоб в ньому багаторазово відбувалась перевірка підрахунку кількості правильних відповідей?

В умові наступного завдання використано поняття *паліндром* (від грецьких πάλιν — назад, знов, та δρόμος — біг) — слово або віршований рядок, що однаково читають в обох напрямках (зліва направо та справа наліво).

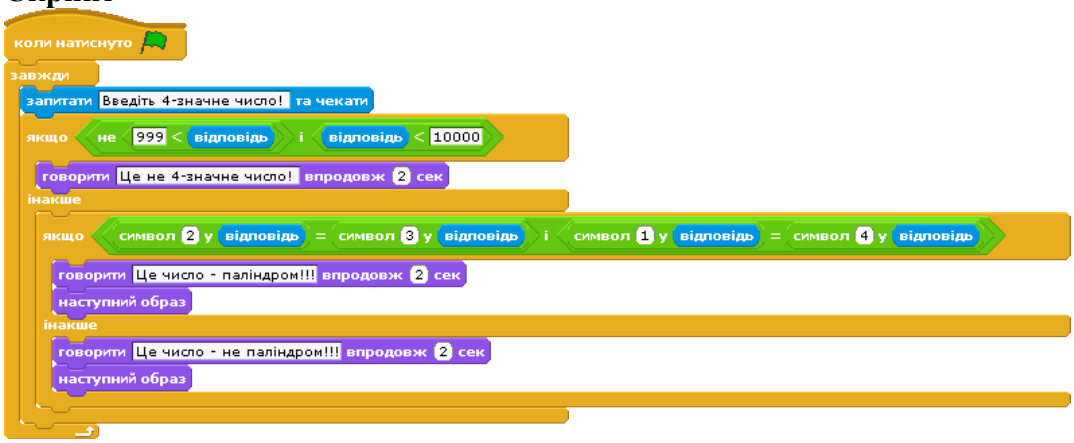
Завдання 3. Старий Кажан дізнався, що таке паліндром. Допоможіть нашому герою навчився визначати, чи є задане чотиризначне число паліндромом.

Алгоритм виконання

Ввести чотиризначне число.

Перевірити коректність введених данних (числа).
Перевірити, чи є задане число паліндромом. Інакше кажучи, визначити, чи збігаються перша з четвертою, а друга з третьою. Ці дві умови мають справджуватися *одночасно*.

Скрипт



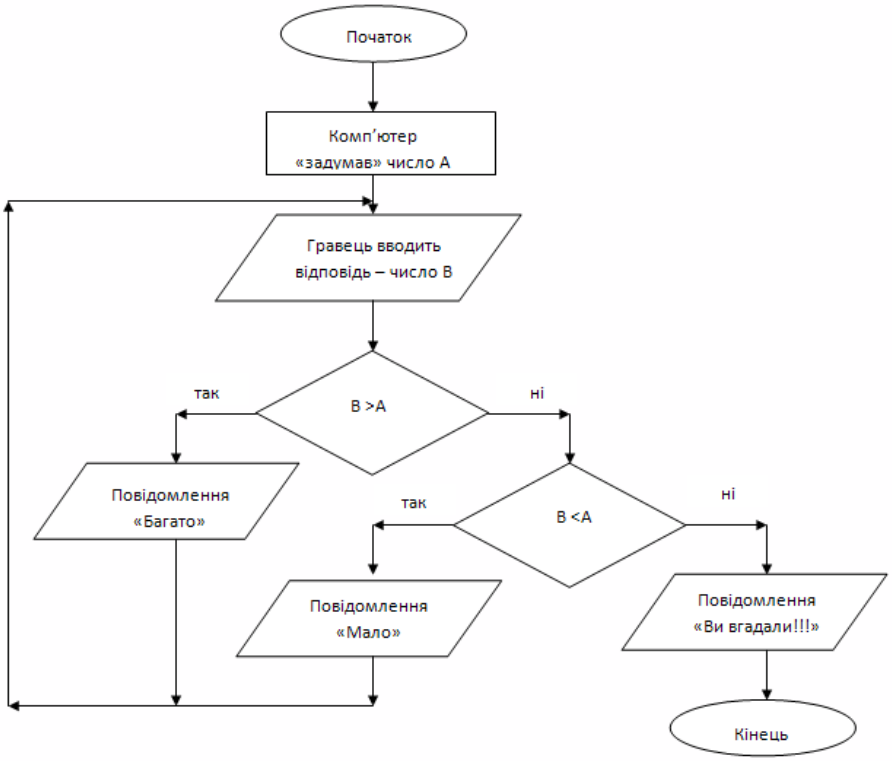
Запустити проект на виконання в неповноекранному режимі.

Дайте відповіді на запитання

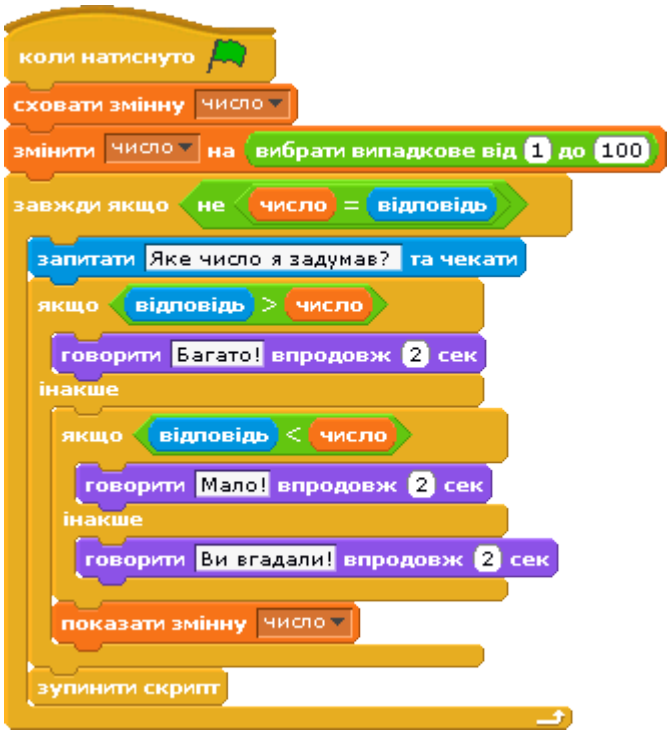
Як реагує програма, коли користувач вводить числа з більшою або меньшою кількість цифр?
Які зміни необхідно внести до скрипта проекту, щоб відбувалась перевірка 5-значних (6-значних) чисел?
А якщо в числі N цифр?
Які зміни потрібно внести, щоб змінну кількість цифр користувач задавав сам?

Завдання 4. Комп'ютер випадковим чином породжує число у межах від 0 до 100, а гравець вгадує його. На пропозиції гравця комп'ютер повідомляє: «Мало», «Багато» чи «Ви вгадали!!!» залежно від взаємного розташування числа-здогадки і випадкового числа.

Блок схема алгоритму



Скрипт



Запускаємо проект на виконання, натиснувши на зелений прапорець у правому верхньому куті програмного вікна).

5. Підбиття підсумків уроку
Виставлення оцінок.

6. Домашнє завдання