



SCENARIUSZ LEKCJI POKAZOWEJ

opracowany w ramach projektu pt. „Szkoła Ćwiczeń w Gminie Rawicz”

Nr i obszar przedmiotowy	Część VI- obszar nauczania TIK
Nazwa przedmiotu	Informatyka
Poziom nauczania	Klasy IV-VIII szkoły podstawowej
Liczba godzin lekcyjnych	2 godziny
Klasa	V
Imię i nazwisko Autora/-ki/Autorów	Mariusz Kwiatkowski
Nazwy szkoły:	Szkoła Podstawowa im. Janusza Korczaka w Sierakowie
Temat lekcji:	Tworzymy grę dla jednego gracza w języku Scratch.

I. **Wstęp do scenariusza (wprowadzenie merytoryczne):**

Współczesna szkoła ma stwarzać uczniom warunki do nabywania wiedzy i umiejętności potrzebnych do rozwiązywania problemów z wykorzystaniem metod i technik wywodzących się z informatyki związanych z rozwojem myślenia komputacyjnego, a w tym logicznego i algorytmicznego myślenia, programowania, posługiwania się aplikacjami komputerowymi, wyszukiwania i wykorzystywania informacji z różnych źródeł, posługiwania się komputerem i podstawowymi urządzeniami cyfrowymi oraz stosowania tych umiejętności na zajęciach z różnych przedmiotów m.in. do pracy nad tekstem, wykonywania obliczeń, przetwarzania informacji i jej prezentacji w różnych postaciach. Od wielu lat komputery wywierają coraz większy wpływ na zmiany zachodzące w funkcjonowaniu społeczeństw: w gospodarce, administracji, bankowości, handlu, komunikacji, nauce i edukacji, czy życiu osobistym obywateli. Informatyka jako dziedzina wiedzy wraz z technologiami, które wspiera, integruje się z niemal wszystkimi innymi dziedzinami i staje się ich nieodłącznym elementem. Wczesny kontakt z informatyką w szkole powinien przybliżyć uczniom możliwości zastosowań tej dziedziny oraz wzbudzić



zainteresowanie informatyką.

Oczekuje się, że wkraczający w zawodowe i dorosłe życie uczniowie będą przygotowani do podjęcia obowiązków i wyzwań, jakie stawia przed nimi XXI wiek. Powinni zatem poznać podstawowe metody informatyki, aby w przyszłości stosować je w praktycznych sytuacjach w różnych dziedzinach. Do tej pory dużą uwagę w edukacji przywiązywano do kształcenia umiejętności korzystania z aplikacji komputerowych oraz zasobów i komunikacji w sieci, obejmując wszystkich uczniów kształceniem w zakresie technologii informacyjno-komunikacyjnej. Oczekiwane obecnie kompetencje obywateli w zakresie technologii cyfrowej wykraczają poza tradycyjnie rozumianą alfabetyzację komputerową i biegłość w zakresie korzystania z technologii. Te umiejętności są nadal potrzebne, ale nie są już wystarczające w czasach, gdy informatyka staje się powszechnym językiem niemal każdej dziedziny i wyposaża je w nowe narzędzia. Podstawowe zadanie szkoły – alfabetyzacja w zakresie czytania, pisania i rachowania – wymaga poszerzenia o alfabetyzację w zakresie umiejętności rozwiązywania problemów z różnych dziedzin ze świadomym wykorzystaniem metod i narzędzi wywodzących się z informatyki oraz na lepsze zrozumienie, jakie są obecne możliwości technologii, komputerów i ich zastosowań. Elementem powszechnego kształcenia staje się również umiejętność programowania. Programowanie jest tu rozumiane znacznie szerzej niż tylko samo napisanie programu w języku programowania. To cały proces, informatyczne podejście do rozwiązywania problemu: od specyfikacji problemu (określenie danych i wyników, a ogólniej – celów rozwiązania problemu), przez znalezienie i opracowanie rozwiązania, do zaprogramowania rozwiązania, przetestowania jego poprawności i ewentualnej korekty przy użyciu odpowiednio dobranej aplikacji lub języka programowania. Tak rozumiane programowanie jest częścią zajęć informatycznych od najmłodszych lat, wpływa na sposób nauczania innych przedmiotów, służy właściwemu rozumieniu pojęć informatycznych i metod informatyki. Wspomaga kształcenie takich umiejętności jak: logiczne myślenie, precyzyjne prezentowanie myśli i pomysłów, sprzyja dobrej organizacji pracy, buduje kompetencje potrzebne do pracy zespołowej i efektywnej realizacji projektów. Umiejętności nabyte podczas programowania są przydatne na zajęciach z innych przedmiotów, jak i później w różnych zawodach, niekoniecznie informatycznych.



II. Obszar tematyczny/problemowy:

Programowanie - Kompozycje, historyjki i gry w języku Scratch

III. Zagadnienie metodyczne stanowiące podstawę przygotowania lekcji / cele dla praktykanta/młodego nauczyciela w zakresie rozwijania kompetencji metodycznych / kompetencji kluczowych

- Utrwalenie i rozszerzenie wiedzy z działu „Programowanie” dotyczącej zasad tworzenia programów komputerowych przy użyciu programu edukacyjnego Scratch.
- Nauka myślenia przyczynowo skutkowego – realizowana poprzez tworzenie własnych rozwiązań problemów
- Kształtowanie samodzielności, sprawności i odpowiedzialności za własne działania – realizowane poprzez samodzielny wybór pracy oraz umiejętność współpracy w grupie.
- Towarzyszenie uczniowi w procesie tworzenia algorytmu programu – poprzez obserwację pracy ucznia, stosowanie techniki zadawania pytań, udzielanie ustnej informacji zwrotnej.
- Sposoby pozyskiwania informacji zwrotnej – pogadanka, obserwacja pracy uczniów, ćwiczenia, wykorzystanie arkusza obserwacji, podsumowanie zajęć z wykorzystaniem samooceny pracy, metody zdań niedokończonych (co również staje się sposobem na weryfikację pracy i osiągnięć uczniów). Informacja zwrotna może zostać wykorzystana do dalszego planowania oraz ewaluacji pracy na zajęciach.
- Indywidualizacja pracy – poprzez przygotowanie zestawu ćwiczeń dostosowanych do poziomu, potrzeb i umiejętności ucznia,
- Nauka planowania, organizacji czasu i miejsca pracy ucznia – realizowana poprzez samodzielny wybór metody i kolejności rozwiązania zadania, działania zgodne lub nie zgodne z wybraną instrukcją, gospodarowanie czasem.

IV. Treści nauczania/uczenia się oraz wskazanie, jakie kompetencje kluczowe uczniowie kształcą podczas lekcji:

Realizowany na zajęciach zakres treści jest zgodny z podstawą programową MEN dla drugiego etapu edukacyjnego.
Ćwiczenia i przykłady wykorzystane na zajęciach stanowią zbiór propozycji do realizacji tematu: „Tworzymy grę dla jednego gracza w języku Scratch” zawartego w podręczniku do informatyki Grażyny Koba.



Materiały są o różnym poziomie trudności, dają możliwość wyboru przez ucznia sposobu pracy (samodzielne lub z pomocą). Dzieci bardziej uzdolnione, jak i dysfunkcjami w uczeniu się mają możliwość satysfakcjonującej dla siebie nauki.

1. Zakres treści w dziale „Programowanie”:

- a) Programujemy w środowisku Baltie - proste animacje.
- b) Tworzymy złożone animacje.
- c) Ustalamy kolejność odtwarzania animacji.
- d) Sztuczki w programie Baltie i zadania.
- e) Programujemy w języku Scratch.
- f) Umieszczamy duszka w określonym miejscu sceny i stosujemy powtarzanie poleceń.
- g) Programujemy historyjkę.
- h) Tworzymy grę dla jednego gracza.
- i) Stosujemy zmienne – zliczamy punkty.
- j) Zadania projektowe – Baltie.
- k) Zadania projektowe – Scratch.

2. Treści nauczania w odniesieniu do podstawy programowej dla drugiego etapu edukacyjnego.

- a) Pisanie prostego programu przy użyciu edukacyjnego języka programowania.
- b) Sterowanie obiektem graficznym na ekranie, np. w przód, w lewo, w prawo, zmiana położenia obiektu o dowolny kąt.
- c) Określanie problemu i celu do osiągnięcia.
- d) Analiza sytuacji problemowej, opracowanie rozwiązania, sprawdzenie rozwiązania problemu dla przykładowych danych, zapisanie rozwiązania w postaci programu.
- e) Projektowanie, tworzenie i zapisywanie w wizualnym języku programowania pomysłów historyjek.
- f) Stosowanie poleceń iteracyjnych i warunkowych oraz zdarzeń.
- g) Zapisywanie poleceń składających się na sterowanie obiektem na ekranie.
- h) Testowanie na komputerze programu pod względem zgodności z przyjętymi założeniami. Modyfikowanie programów.
- i) Objaśnianie przebiegu działania programów.

3. Treści i zagadnienia nie wymienione w punkcie 2 wychodzą ponad podstawę



programową przewidzianą dla edukacji na poziomie 2.

V. Cele ogólne lekcji (kierunki dążeń pedagogicznych w obszarze wiadomości, umiejętności, postaw)

1. Pokazanie, na czym polega programowanie, w tym realizacja sytuacji warunkowych oraz stosowanie zmiennych.
2. Tworzenie programów w języku Scratch.
3. Rozwijanie kreatywności i kształcenie logicznego myślenia.

VI. Cele ucznia w zakresie kształtowania wiedzy, umiejętności, postaw i systemu wartości:

1. Poznanie sposobów rozwiązywania problemów z wykorzystaniem komputera.
2. Zapoznanie się z możliwościami sterowania obiektem graficznym na ekranie.
3. Poznanie zasad tworzenia programu komputerowego przy użyciu programu edukacyjnego. Rozwijanie zdolności logicznego myślenia.
4. Kształtowanie umiejętności samodzielnego rozwiązywania zadań problemowych.
5. Uświadomienie korzyści ze stosowania właściwego oprogramowania (tu programu edukacyjnego) dla własnego rozwoju.

VII. Metody pracy z uczniami:

Metoda eksponująca, metoda problemowa i metoda ćwiczeń praktycznych

VIII. Środki dydaktyczne (wykorzystane przez uczniów oraz przez nauczyciela):

Sala wyposażona w projektor multimedialny i ekran, komputery z zainstalowanym oprogramowaniem Scratch.

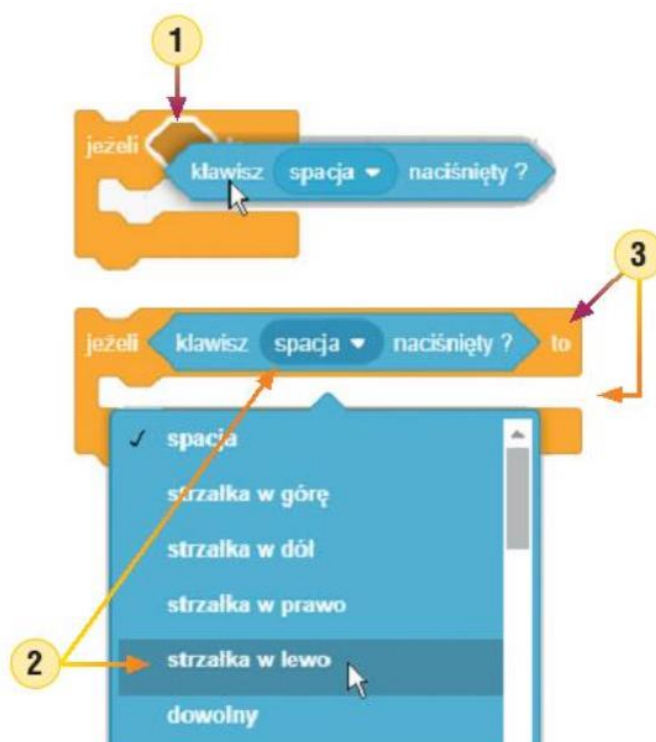
IX. Przebieg lekcji, zawierający co najmniej¹: wprowadzenie do lekcji, podejmowane działania, sposoby weryfikacji założonych celów lekcji:

- 1) Nauczyciel sprawdza wiadomości z poprzedniej lekcji i zadaje pytania o poznane polecenia sekwencyjne i iteracyjne oraz sposoby wykonywania operacji sterowania

¹ Zalecane jest również uwzględnienie materiałów i zadań rozszerzających (dla uczniów zdolnych, zainteresowanych daną tematyką oraz uczniów ze specyficznymi trudnościami w uczeniu się).

postać (duszkciem) na scenie w środowisku Scratch.

- 2) Wybrani uczniowie odpowiadają na zadane pytania.
- 3) Nauczyciel w miarę potrzeby uzupełnia odpowiedzi jeżeli nie są pełne lub poprawne.
- 4) Nauczyciel podaje temat i cel lekcji – utworzenie gry dla jednego gracza.
- 5) Wyjaśnia uczniom, na czym ma polegać gra, którą będą tworzyć.
- 6) Korzystając z projektora i przykładu 4. (str. 89) podręcznik,



nauczyciel omawia stosowanie polecenia warunkowego w wersji uproszczonej w języku Scratch.

Nauczyciel poleca zapisanie jako notatki:

Aby w języku Scratch zrealizować sytuację warunkową, możemy użyć polecenia „jeżeli” z grupy Kontrola

- 7) Uczniowie wykonują ćwiczenia 11 i 12 (str. 89-90), korzystając z przykładu 4. (str. 89) i metody przedstawionej na str. 89. Nauczyciel sprawdza wykonywanie ćwiczeń i pomaga nieradzącym sobie uczniom.
- 8) Jeżeli uczniowie sprawnie wykonają ćwiczenia 11 i 12 nauczyciel może wyjaśnić im tworzenie zmiennych przykład 6 str.91 (str. 91) i zapisem skryptu zliczającego punkty przykład 7 (str. 91) (zadanie dla uczniów zdolnych lub na kolejne zajęcia



rozszerzone o dodanie kolejnych postaci ćwiczenia 13 i 14 str. 92).

- 9) W podsumowaniu zajęć wybrani uczniowie prezentują rozwiązania ćwiczeń 11. i 12. (str. 89-90).

X. Podsumowanie:

W temacie tym uczniowie będą tworzyć program- grę dla jednego gracza za pomocą środowiska programowania Scratch. Będzie to program, w którym uczniowie powinni się nauczyć stosowania instrukcji warunkowej do sterowania ruchem postaci (duszka).

Wprowadzanie polecenia warunkowego należy dokonać z właściwą starannością i zacząć od zapoznania ze sposobem stosowania warunku w wersji uproszczonej. Do realizacji warunku uproszczonego zastosować należy pętlę **Jeżeli ... to** w której warunkiem jest wybrane zdarzenie z grupy Czujniki: **klawisz spacja naciśnięty?** Należy wyjaśnić uczniom, że aby sterować postacią (duszką) we wszystkich kierunkach to warunek musi być skopiowany (cztery pętle i w każdej w czujce zastosowany inny klawisz). Wszystkie cztery pętle **jeżeli ... to** muszą być umieszczone wewnątrz pętli **zawsze**.

Na kolejnej lekcji uczniów należy zapoznać, z tworzeniem zmiennych i wykorzystaniem ich do zliczania punktów. Dla niektórych uczniów może to być trudne zagadnienie.

Aby uczniowie dobrze radzili sobie z wykonywaniem ćwiczeń i zapisem skryptu programu, należy dokładnie wyjaśniać przykłady. Jeśli uczniowie na początku rozumieją poszczególne zasady zapisu warunku, deklaracji zmiennych oraz tworzeniem skryptów będą sprawniej i z większym zaangażowaniem wykonywać kolejne, nawet trudniejsze ćwiczenia i zadania.

XI. Literatura:

Informatyka dla Szkoły Podstawowej „Teraz bajty” klasa V , Grażyna Koba, wyd. Migra

[Materiały pomocnicze](#)

[Podręcznik Scratch](#)

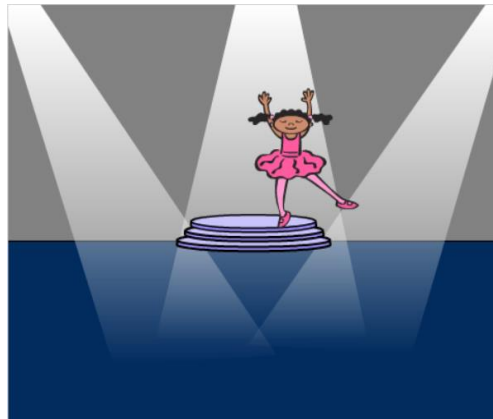
Załączniki do scenariusza (np. karty pracy, zestawy ćwiczeń dla uczniów, teksty źródłowe, ilustracje):

1. Załącznik nr 1: Zestaw ćwiczeń

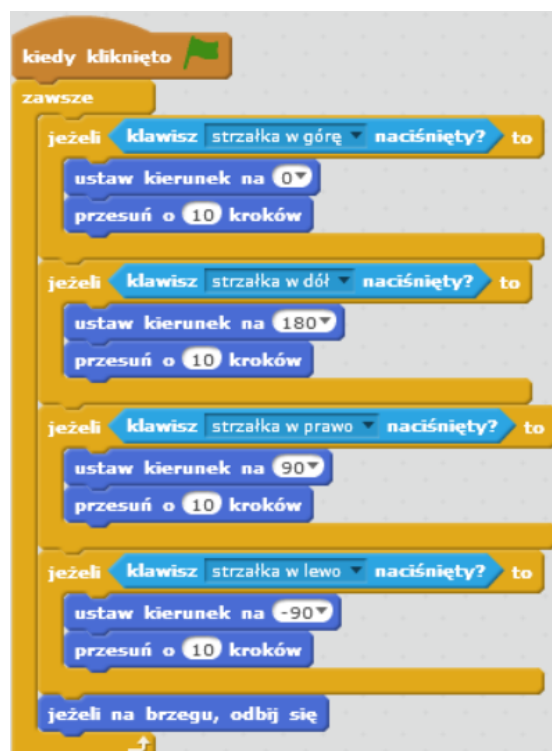


Ćwiczenie 11. Sterujemy duszkiem w czterech kierunkach

1. Utwórz program, w którym scena zostanie zmieniona na wybrany obraz, a duszek **balerina** będzie się poruszał po scenie w prawo, w lewo, w górę, w dół zależnie od naciśniętych strzałek.



2. Wszystkie polecenia powtarzaj nieskończenie wiele razy.



3. Zapisz program w pliku pod nazwą: **balerina z piłką**.

Wskazówki:

- Tło sceny **spotlight-stage2** zostało wybrane z gotowych tła programu Scratch.
- Na rysunku skryptu powyżej przedstawione polecenia można duplikować. Po zapisaniu pierwszej pętli dla zmiany kierunku w górę (za pomocą strzałki w górę) duplikujemy pierwszej pętli dla zmiany kierunku w górę (za pomocą strzałki w górę) duplikujemy



pętlę dla pozostałych kierunków i zmieniamy odpowiednio klawisz (strzałka) w czujniki i kierunek ruchu w poleceniu **ustaw kierunek na** odpowiedni kierunek(0-góra, 180-dół, 90-prawo, -90-lewo).

Ćwiczenie 12. Piszemy skrypt dla duszka **piłka** poruszającego się losowo po całej scenie

1. Otwórz plik **balerina z piłką** zapisany w ćwiczeniu 11.
2. Dodaj duszka-**piłka** i utwórz dla niego skrypt, w którym duszek będzie poruszał się nieskończenie wiele razy po całej scenie.

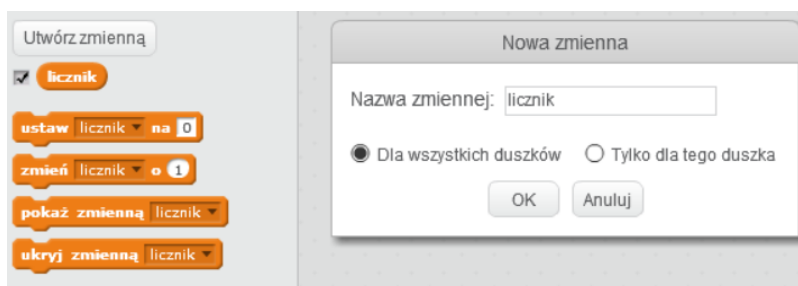


3. Zapisz plik pod tą samą nazwą.

Wskazówka: Zastosuj polecenie **leć** i umieść je w pętli **zawsze**.

Ćwiczenie 13. Zliczamy punkty w grze

1. Otwórz plik **balerina z piłką** zapisany w ćwiczeniu 12 i zmodyfikuj program tak, aby gracz otrzymywał jeden punkt, gdy duszki się dotkną.
2. Utwórz zmienną o nazwie licznik:



- W grupie **Dane** na karcie **Skrypty** kliknij przycisk **Utwórz zmienną** – otworzy się okno dialogowe **Nowa zmienna**.
- W polu tekstowym **Nazwa zmiennej** wpisz nazwę zmiennej np. licznik.



- Kliknij przycisk **OK**.
- Utworzona zmienna i polecenia dotyczące jej stosowania pojawią się w panelu poleceń w grupie **Dane**.

3. Uzupełnij skrypt duszka-piłka.



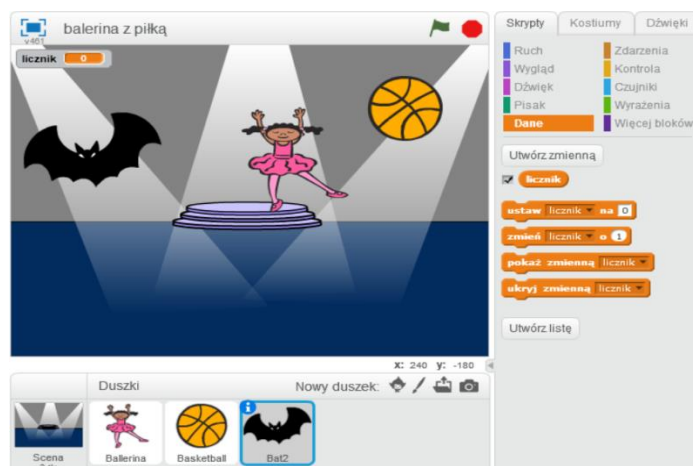
4. Zapisz plik pod ta sama nazwą.

5. Uruchom program i obserwuj na scenie zmieniającą się wartość zmiennej licznik.

Uwaga: Licznik po wykonaniu polecenia **ustaw licznik na 0** jest równy zero. Wykonanie pierwszy raz polecenia **zmień licznik o 1** spowoduje, że licznik przyjmie wartość 1. kolejne wykonanie polecenia **zmień licznik o 1** powoduje, że licznik zwiększa się o jeden, czyli będzie przyjmował wartości: 2, 3, 4 itd.

Ćwiczenie 14. Dodajemy trzeciego duszka i wprowadzamy odejmowanie punktów

1. Otwórz plik zapisany w ćwiczeniu 13.
2. Dodaj kolejnego duszka- nietoperza, który ma się poruszać losowo po całej scenie, podobnie jak piłka.





3. Wprowadź w skrypcie duszka-nietoperza warunek odejmowania punktów w grze – gdy duszek-balerina i duszek-nietoperz dotkną się, licznik ma zmniejszyć się o 1.



4. Zapisz program w pliku pod nazwą gra trzy duszki.

Wskazówka: Polecenia dla duszka-nietoperza są bardzo podobne do poleceń duszka-piłki, dlatego można skopiować polecenia duszka piłki i przeciągnąć je na duszka nietoperza w obszarze Duszki. Następnie wprowadzić należy niezbędne poprawki i usunąć zerowanie zmiennej licznik w skrypcie duszka-nietoperza.