



Elektroniczna kostka do gry

Nauka kodowania oparta o wykorzystanie zjawisk przyrodniczych

Autor: Tomasz Sobiepan

Przedmiot: Informatyka

Klasa: IV - V

Czas: 1 x 45 minut lub 2 x 45 minut

Cele zajęć. Uczeń:

- 1) zapozna się eksperymentalnie z pojęciem przyspieszenia, wskazywanym przez akcelerometr;
- 2) zastosuje następujące elementy programowania: pętle i różne rodzaje ich przerwań, instrukcje warunkowe, zagnieżdżanie instrukcji, generator liczb losowych, programowanie matrycy diod LED, wartość bezwzględna liczby;
- 3) ułoży program wykonujący wskazane zadania;
- 4) zaproponuje i wykorzysta w praktyce swój własny, pomysł na alternatywne skonstruowanie programu.

Metody dydaktyczne

- **Nauka programowania oparta o wykorzystanie zjawisk przyrodniczych**

To nowoczesna metoda nauczania, pozwalająca na wdrożenie modelu STEM na lekcjach każdego przedmiotu. Dzięki wykorzystaniu zjawisk przyrodniczych do nauki programowania oraz kodowania do nauki przedmiotów przyrodniczych, zapewnia międzyprzedmiotową spójność wiedzy i umiejętności uczniów oraz uatrakcyjni naukę obu tych dziedzin.

- **IBL (ang.: Inquiry Based Learning)** – uczenie się przez dociekanie.

Formy pracy

- Praca własna uczniów (w grupach 2-3 osobowych):
 - analiza podręcznika
 - eksperymentowanie przyrodnicze
 - programowanie z rozsypanki bloczków
- Praca indywidualna uczniów
- Instruktaż nauczyciela
- Burza mózgów
- Dyskusja

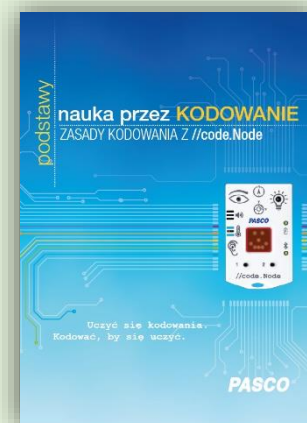
Podstawa programowa

Cele kształcenia – wymagania ogólne

- I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.
- II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Treści nauczania – wymagania szczegółowe (KLASY IV–VI)

- II Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:
 - 1) projektuje, tworzy i zapisuje w wizualnym języku programowania:
 - a) pomysły historyjek i rozwiązania problemów, w tym proste algorytmy z wykorzystaniem poleceń sekwencyjnych, warunkowych i iteracyjnych oraz zdarzeń,
 - b) prosty program sterujący robotem lub innym obiektem na ekranie komputera;
 - 2) testuje na komputerze swoje programy pod względem zgodności z przyjętymi założeniami i ewentualnie je poprawia, objaśnia przebieg działania programów;



Materiały potrzebne na lekcji

- Urządzenie **//code.Node** oraz program SPARKvue.
- Podręcznik „**Nauka przez kodowanie – zasady kodowania z //code.Node**”.
- Kostki do gier planszowych.

Wiedza i umiejętności uczniów, na których bazuje tematyka zajęć

- podstawowe umiejętności konturowania programów w języku Blockly;
- zastosowanie pętli przerywanej warunkiem lub przyciskiem **Zatrzymaj** w doświadczeniu;
- wykorzystanie warunku **jeżeli** o różnych strukturach;
- programowanie matrycy diod LED 5x5 urządzenia **//code.Node**.

Przebieg zajęć

1. Czynności organizacyjne i przedstawienie uczniom celów lekcji.
 - a. Każdy z nas wie, jak posługiwać się kostką używaną w grach planszowych. Jest taka sama z każdej strony, więc gdy rzucimy nią pozwalając, by turlała się przez pewien czas – nigdy nie wiadomo, która z sześciu ścian znajdzie się na górze po zatrzymaniu się kostki. Mówimy, że w ten sposób losujemy jedną z sześciu liczb.
 - b. Aby uzyskać taki sam efekt, ale elektronicznie, programiści używają specjalnych funkcji do losowania liczb z podanego zakresu. Funkcje te nazywają się generatorami liczb losowych i wykorzystywane są w wielu dziedzinach, na przykład w kryptografii do szyfrowania tajnych wiadomości.
 - c. Prawdziwą kostkę do gry wprawiamy w ruch, by uzyskać efekt losowania. Trudno wyobrazić sobie takie postępowanie przy programowaniu (rzucanie komputerem jest bez sensu 😊). Na szczęście możemy wykorzystać urządzenie **//code.Node**, które potrafi wykryć ruch i przekazać informację o nim w formie elektronicznej, możliwej do wykorzystania przy kodowaniu.
 - d. Napijemy więc program, który spowoduje, że po poruszeniu urządzeniem **//code.Node**, na jego wyświetlaczu pojawi się losowo wybrana liczba od 1 do 6. Przy okazji nauczymy się kilku ważnych procedur matematycznych i programistycznych.
2. Podział klasy na grupy 2-3 osobowe. Każda otrzymuje kostkę do gry, urządzenie **//code.Node**, komputer lub tablet oraz podręcznik. Uczniowie wykonują polecenia przygotowawcze:
 - a. załadujcie (w sposób wskazany przez nauczyciela) plik *kostka_do_gry_i_wykresy.spklab*, na przykład ze strony <https://pasco.com.pl/biblioteka-kodow-blockly/>;
 - b. włączcie urządzenie **//code.Node** i połączcie je przez Bluetooth z komputerem (tabletem).
3. Pobawcie się i sprawdźcie, jak to działa:
 - a. rozpocznijcie działanie programu przyciskiem **Start**;
 - b. potrząśnijcie urządzeniem **//code.Node** i obserwujcie jego wyświetlacz (matrycę diod LED 5x5), porównajcie działanie „elektronicznej” i zwykłej kostki do gry;
 - c. po skończonej zabawie przerwijcie działanie kodu przyciskiem **Zatrzymaj**, ale nie wyłączajcie programu SPARKvue.
4. Aby samodzielnie napisać taki program, jakim się bawiliście, trzeba poznać pewne własności czujnika i języka programowania Blockly. Zapoznajcie się z informacjami w podręczniku (rozdział „**Badanie prawdopodobieństwa i ruchu**”). Zwróćcie uwagę na wymienione poniżej zagadnienia.

- a. Jak działa czujnik ruchu w urządzeniu **//code.Node**?
Jego wskazania podczas Waszej zabawy możecie obejrzeć w programie SPARKvue, gdy przejście do strony doświadczenia, która nazywa się „Wykresy”. Jeśli włączycie program przyciskiem **Start**, możecie zobaczyć na bieżąco (czyli w czasie rzeczywistym) wykresy przyspieszenia w kierunku X i w kierunku Y, rejestrowanego przez **//code.Node**.
- b. Generator liczb losowych – co to jest i jakie ma zastosowanie?
- c. Jaka jest zasada działania matrycy diod LED 5x5 w urządzeniu **//code.Node**?
- d. Co oznacza w matematyce pojęcie wartość bezwzględna liczby?

5. Jak zaprogramować wyzwalacz uruchamiany potrząśnięciem urządzenia **//code.Node**? Generowanie przez uczniów pomysłów, zapisywanych na tablicy, kierowane przez nauczyciela. Przewidywane ustalenia:

☞ Podczas pracy uczniowie mogą włączyć doświadczenie i obserwować wykresy przyspieszenia, by na bieżąco weryfikować zgłaszane pomysły.

- a. rolą wyzwalacza jest przekazanie sygnału do działania dla dalszej części programu, gdy wydarzy się oczekiwana sytuacja (u nas – gdy potrząśniemy urządzeniem **//code.Node**);
 - b. program musi brać pod uwagę wskazania przyspieszenia rejestrowane przez czujnik;
 - c. w zależności od sposobu potrząśnięcia, czasami większe będzie **Przyspieszenie X**, a innym razem **Przyspieszenie Y**;
 - d. aby nasz wyzwalacz działał za każdym razem, musimy uwzględnić oba przyspieszenia;
 - e. dodanie przyspieszeń nie jest dobrym rozwiązaniem, bo mogą one przyjmować wartości zarówno dodatnie, jak i ujemne, więc te wartości mogą się „znosić”;
 - f. rozwiązaniem jest dodawanie wartości bezwzględnych przyspieszeń;
 - g. skonstruowanie wyzwalacza wymaga użycia funkcji **jeśli**, która sprawdza wartość dodanych wartości bezwzględnych przyspieszeń;
 - h. wyzwalacz nie może ciągle generować sygnałów do działania – powinien poczekać, aż przestaniemy poruszać urządzeniem **//code.Node**;
 - i. „czekanie” wyzwalacza możemy zrealizować za pomocą warunku **jeżeli**, ale lepszym pomysłem jest utworzenie pętli zawierającej bloczek oczekiwania, przerywanej małą wartością sumy wartości bezwzględnych przyspieszeń.
6. Programowanie matrycy. Uczniowie zapisują na kartkach wzory poszczególnych liczb z kostki do gry, jakie trzeba zaprogramować na matrycy diod LED 5x5. Można przeprowadzić to tak samo, jak odpowiedni fragment lekcji „Urządzenie do rozróżniania biegunów magnetycznych”.
7. Rozsypanka. Grupy uczniów na chwile zamieniają się miejscami tak, że każda siada przy stanowisku innej grupy. Ich zadaniem jest porozdzielać i pomieszać bloczki z programu na ekranie (nie wolno żadnego z nich skasować). Następnie grupy wracają na swoje miejsce, a ich zadaniem jest ułożyć poprawnie działający program elektronicznej kostki do gry. Nauczyciel może, wedle uznania, wprowadzić element rywalizacji – wykonanie zadania na czas.
8. Finał – nauczyciel gratuluje grupom sukcesu i pyta, jak pokonywały trudności oraz podsyca poczucie zadowolenia i satysfakcji uczniów.
9. Podsumowanie – uczniowie odpowiadają w dowolnej formie (ustnej, pisemnej, elektronicznie) na dwa pytania:

- a. Czego ważnego nauczyłam(-em) się na dzisiejszej lekcji? Odpowiedź może dotyczyć zarówno zjawisk przyrodniczych, jak i umiejętności programowania.
- b. Który z problemów był dla mnie najtrudniejszy i co spowodowało, że jednak poradziłam(-em) sobie z nim?

Zadanie domowe

Programowanie alternatywne. Dokonać jednej modyfikacji kodu tak, by działał on identycznie, jak na lekcji.

Uczniowie mogą korzystać w domu z bezpłatnej wersji programu SPARKvue na urządzenia mobilne (wersja ta jest taka sama pod względem wizualnym i funkcjonalnym, co wersja na komputery). Mogą też przestać sobie mailiem program napisany na lekcji.

Odpowiedź: modyfikacja może dotyczyć na przykład zastosowania innego bloczka programowania matrycy diod LED 5x5 lub zastosowania warunku **jeśli** do oczekiwania na koniec potrząśnięcia czujnikiem.

Uwaga

Zamiast zadawania pracy domowej można temu zadaniu poświęcić następną lekcję.

Rozszerzenie

Czujniki PASCO można także zaprogramować używając języka tekstowego Python, ale jest to zadanie na późniejszy etap nauczania informatyki.

