

Robo-matematycy

Autorki: Joanna Płatkowska-Nęcka, Karolina Czerwińska

Lekcja 2:

Silniki, głośniki i LEDy, czyli poznajemy się bliżej z mBotem2

Lekcja, podczas której uczniowie oswoją się z programowaniem mBota2, wykorzystując jego LEDy, głośnik i silniki. Dowiedzą się, ile różnych kolorów światła może wyświetlić robot. Podejmą także próbę muzykowania przy użyciu mBota2.

Cele zajęć:

Uczeń powinien:

- wykorzystać informacje dotyczące części robota do zaprogramowania ruchu,
- wykorzystać światła robota do pokazania flag wybranych państw lub "opowieści kolorami",
- stworzyć krótką melodię graną przez robota,
- stworzyć krótką grę, która wykorzystuje wcześniej poznane funkcjonalności Scratcha i mBota2.

Materiały pomocnicze:

- roboty mBot2
- program mBlock (do kodowania robota) w wersji online <https://ide.mblock.cc/> lub offline <http://www.mblock.cc/download/>
- komputery stacjonarne lub laptopy (ze sprawnym portem USB)
- laptop lub komputer nauczyciela podłączony do tablicy multimedialnej lub projektora

Pojęcia kluczowe:

→ robot → silnik → RGB → LED → głośnik → mikrokontroler CyberPi → mBlock

Czas na realizację zajęć: 45 minut (1 godzina lekcyjna)

Metody pracy:

- grupowa - praca w grupach z robotem,
- ćwiczenia praktyczne.

Treści programowe (związek z podstawą programową)

Podstawa programowa kształcenia ogólnego dla szkół podstawowych – II etap edukacyjny – klasy IV-VI, informatyka:

- I. Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:
 2. formułuje i zapisuje w postaci algorytmów polecenia składające się na:
 - 3) sterowanie robotem lub obiektem na ekranie;
 3. w algorytmicznym rozwiązywaniu problemu wyróżnia podstawowe kroki: określenie problemu i celu do osiągnięcia, analiza sytuacji

problemowej, opracowanie rozwiązania, sprawdzenie rozwiązania problemu dla przykładowych danych, zapisanie rozwiązania w postaci schematu lub programu.

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:

1. projektuje, tworzy i zapisuje w wizualnym języku programowania:

- 1) pomysły historyjek i rozwiązania problemów, w tym proste algorytmy z wykorzystaniem poleceń sekwencyjnych, warunkowych i iteracyjnych oraz zdarzeń,
- 2) 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;

IV. Rozwijanie kompetencji społecznych. Uczeń:

1. uczestniczy w zespołowym rozwiązaniu problemu posługując się technologią taką jak: poczta elektroniczna, forum, wirtualne środowisko kształcenia, dedykowany portal edukacyjny;
2. identyfikuje i docenia korzyści płynące ze współpracy nad wspólnym rozwiązywaniem problemów;

V. Przestrzeganie prawa i zasad bezpieczeństwa. Uczeń:

1. posługuje się technologią zgodnie z przyjętymi zasadami i prawem; przestrzega zasad bezpieczeństwa i higieny pracy;

Przebieg zajęć:

1. Wprowadzenie w tematykę i integracja grupy

Nauczyciel tłumaczy uczniom, że na dzisiejszych zajęciach napiszą wspólnie kilka programów, w których wykorzystane będą różne funkcjonalności mBota2. Prosi uczniów o uruchomienie programu mBlock, podłączenie modemu Bluetooth Dongle komputera i o sparowanie robota z komputerem.

2. Część zasadnicza

Zadanie 1: programowanie ruchu robota

Nauczyciel poleca uczniom zaprogramować robota tak, by przejechał z punktu A do punktu B, np. od komputera, przy którym pracuje uczeń do biurka nauczyciela, po drodze obracając się kilka razy wokół własnej osi i skręcając kilka razy w prawo i w lewo.

Sytuacją idealną jest, gdy uczniowie samodzielnie znajdą sposób na rozwiązanie tego zadania. Jeśli okaże się, że mają z nim problem, nauczyciel podpowiada, jakie bloczki należy wykorzystać do rozwiązania tego zadania (np. wyświetla je na projektorze). Uczniowie mogą wykonując to zadanie odkryć, że można korzystać z bloczków typu jedź naprzód czy skręcaj lub bloczków sterujących osobno każdym z silników. Równocześnie mogą zauważyć, że przy bloczkach skręcaj w prawo lub skręcaj w lewo obrót zawsze będzie odbywał się po tym samym łuku. Żeby robot skręcał po łagodniejszym, większym łuku trzeba skorzystać z programowania obu silników osobno.

Dobrze, żeby w trakcie tego zadania uczniowie również zbadali jaki odcinek przejdzie robot w określonym czasie przy różnych prędkościach, a także zastanowili się, w jakich sytuacjach lepiej użyć bloczków jedź przez x sekund, a kiedy lepiej wybrać bloczek określający w centymetrach odległość do przejechania. Skrypt mógłby wyglądać następująco (wykorzystano w nim wszystkie opisane wyżej opcje ruchu):



Jeśli uczniowie szybko wykonają zadanie, można dodawać kolejne sekwencje ruch robota porównując przy okazji działanie różnych bloczków.

Poprawnie zaprogramowany robot będzie wykonywał polecenia jazdy z dużą dokładnością, ale ze względu na zmianę podłoża czy niewielką przeszkodę po drodze, trasa może delikatnie ulec zmianie. Warto na taką okoliczność zaprogramować sobie sterowanie z poziomu klawiatury, które pozwoli skorygować końcową pozycję robota czy zatrzymać go, jeśli przez przypadek zacznie odjeżdżać daleko od punktu startowego.



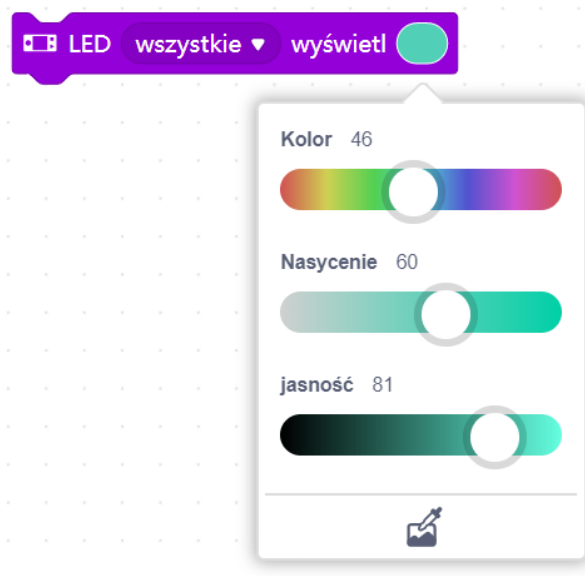
Zadanie 2: programowanie LEDów robota

Nauczyciel zaprasza uczniów do wykonania kolejnego zadania, podczas którego nauczą się wyświetlać na robocie różne kolory. Mówi uczniom, że robot jest wyposażony w pięć diod LED, na których można łatwo wskazać dowolny kolor do wyświetlenia. Warto tu opowiedzieć uczniom, że każdy z wyświetlanych kolorów powstaje z połączenia trzech różnych kolorów. Jest to model RGB. Podpowiada uczniom, że skrót "RGB" pochodzi z angielskiego i oznacza trzy barwy - zachęca uczniów, by zgadli o jakie barwy chodzi (red - czerwony; green - zielony; blue - niebieski)

Nauczyciel wyświetla stronę internetową: <http://www.colorsfire.com/rgb-color-wheel/>. Przesuwając suwakiem, widać, jak zmieniają się wartości trzech modelowych kolorów (RGB) i jakie kolory wówczas powstają. mBlock pozwala oczywiście na wybranie koloru światła bezpośrednio z palety, bez konieczności ustawiania wartości RGB. Pokazanie palety RGB uczniom może jednak pokazać im, w jaki sposób te różne kolory powstają. Uczniowie mogą poeksperymentować i odkryć jakie kolory mogą stworzyć:



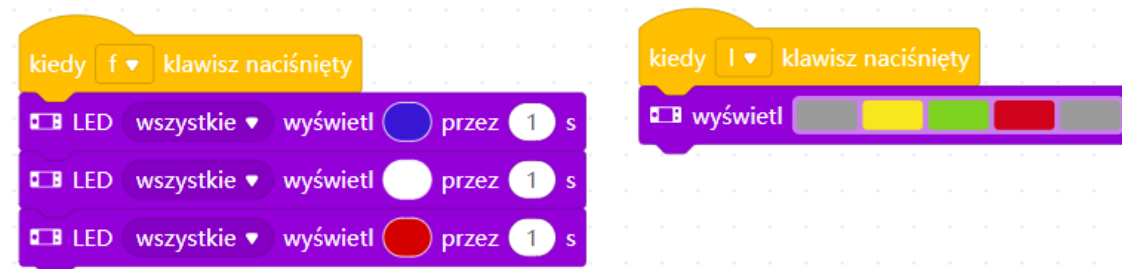
Następnie nauczyciel proponuje uczniom, żeby spróbowali, jakie barwy światła można uzyskać manipulując kolorem, nasyceniem i jasnością:



W programie mBlock można zaprogramować LEDy mBota2 tak, że każda z pięciu diod będzie świecić innym kolorem, a nawet można zaprogramować zmieniające się kolory. Nauczyciel może zaproponować uczniom mniej i bardziej złożoną wersję wykorzystania LEDów w tym zadaniu.

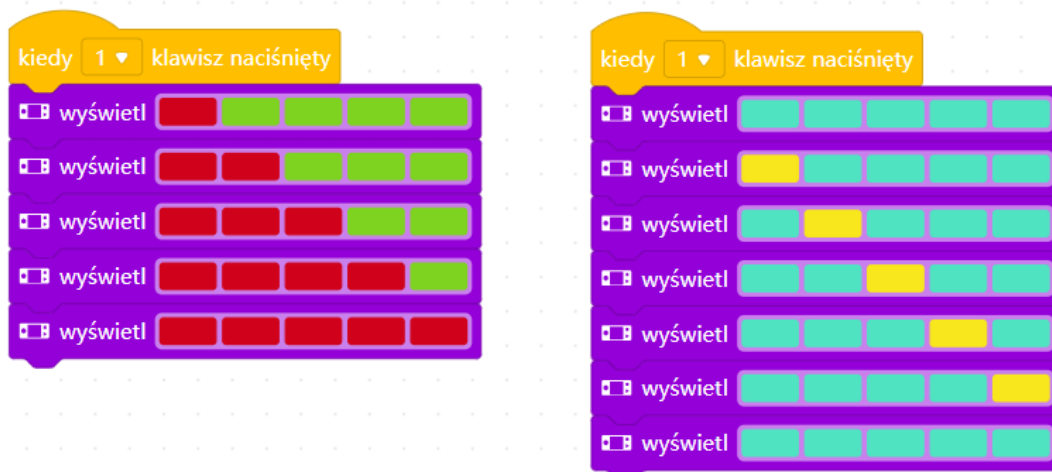
W wersji mniej złożonej nauczyciel zaprasza uczniów do wyszukania flagi dowolnego kraju lub rozdaje uczniom załącznik nr 1 (po jednym wydruku dla każdej z grup). Prosi uczniów, by każda z grup wyświetliła na swoim robocie kolory flagi wybranego państwa. Pozostałe grupy zgadują, o flagę jakiego państwa chodzi. Uczniowie mogą to zrobić na dwa sposoby. W pierwszym wszystkie diody świecą na wybrany kolor przez krótką chwilę, następnie zmieniają się, aż wyświetlą się dwa lub trzy kolory znajdujące się na wybranej fladze. Mogą też wyświetlić te kolory równocześnie wybierając bloczek wyświetl i obok domyślnie kolory tęczy. Po kliknięciu w ten bloczek otworzy się okno, w którym można wybierać kolory podobnie jak w programie Paint. Najpierw klikamy w jeden z dziewięciu kolorów, a następnie w jedną z pięciu diod. Możemy skorzystać z gumki, żeby daną diodę wyłączyć. Po kliknięciu w ikonę dyskietki możemy dany układ zapisać. Po kliknięciu w ikonę trójkąta (play) możemy wyświetlić układy kolorystyczne po kolei.

Poniżej przykład flagi Francji wyświetlony pierwszym sposobem i flagi Litwy wyświetlony drugim sposobem:



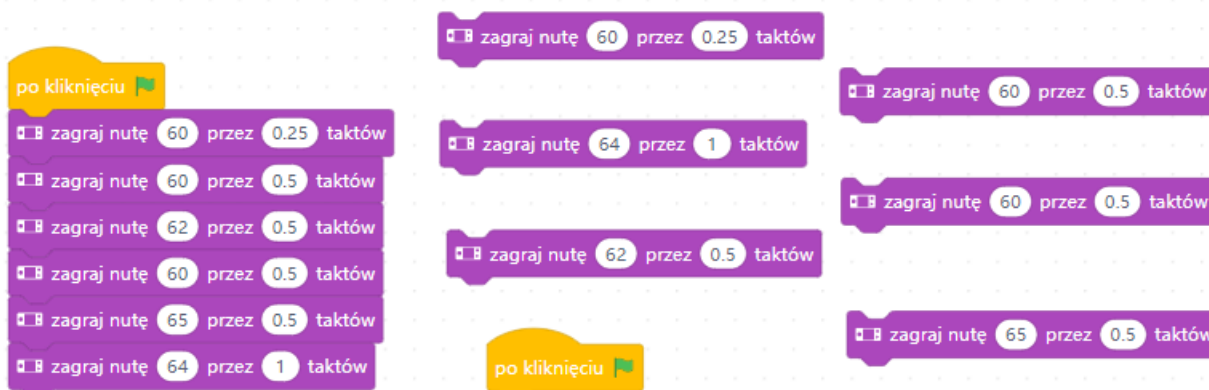
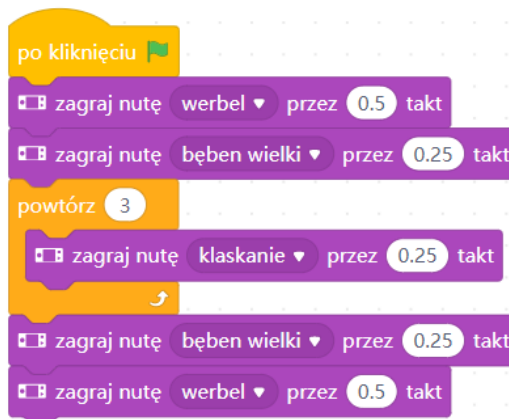
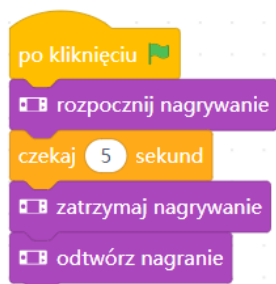
W wersji bardziej złożonej uczniowie mogą spróbować ułożyć "kolorowe historyjki". W tym celu korzystając z bloczka display projektują i zapisują kilka układów kolorystycznych, które kiedy pojawią się jeden po drugim staną się animacją. Uczniowie mogą odgadywać jaką historyjkę chciała zaimować każda z grup.

Poniżej przykład animacji pokazującej pożar lasu oraz rybki przepływającej przez rzekę.



Zadanie 3: programowanie dźwięków

Nauczyciel przypomina uczniom, że mBot2 wyposażony jest w głośnik, a także mikrofon. Wyjaśnia jak można na nim grać różne proste melodie. Zachęca uczniów do testów i wypróbowania dźwięków nut, wbudowanych nagrań robota czy nagrywania własnych dźwięków (mBot2 może zapamiętać i odtworzyć jedno nagranie). Nauczyciel może też umieścić na tablicy rozsypankę, z której uczniowie mogą ułożyć znaną melodię:



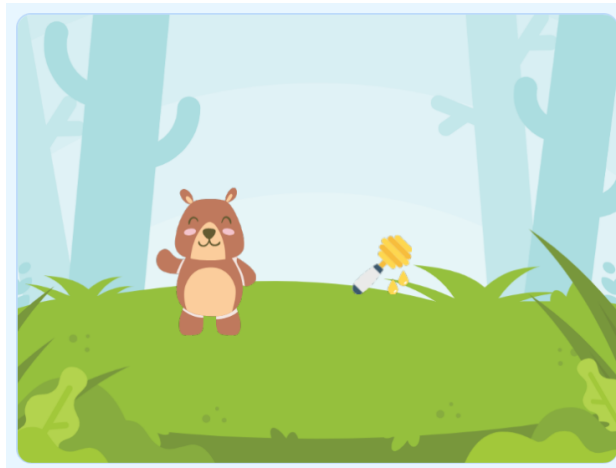
Zadanie 4: włączanie robota do gier – zadanie dodatkowe

Jeśli grupa sprawnie pracuje, nauczyciel proponuje wykonanie dodatkowego zadania, w którym uczniowie napiszą krótką grę komputerową, w którą włączony będzie robot.

Gra polega na tym, że niedźwiedź próbuje złapać pojawiające się w losowo wybranych miejscach łyżeczki z miodem. Za każdym razem, gdy taką łyżeczkę złapie, robot świeci niebieskim światłem. Niedźwiedź porusza się dzięki czujnikowi odległości robota – ręka przyłożona bliżej lub dalej od robota będzie odpowiednio poruszać duszkiem w lewo lub w prawo (tu trzeba zwrócić uwagę, w którą stronę zwrócony jest robot). W przypadku uczniów, którzy dobrze radzą sobie z układem współrzędnych, można zaproponować, żeby miód powoli spadał w różnych miejscach zamiast pojawiać się w losowo wybranych miejscach. Poniżej podano obie wersje skryptu dla miodu.

Oczywiście bohaterami gry mogą być zupełnie inne postaci czy przedmioty i znajdować się w zupełnie innym miejscu.

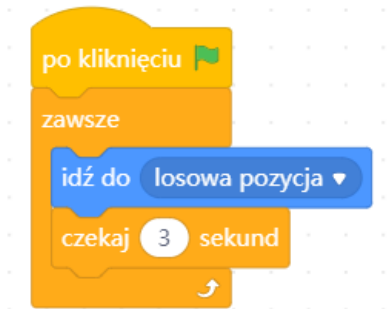
Nauczyciel moderuje dyskusję i zachęca uczniów do zaproponowania, jak może być napisany skrypt gry.



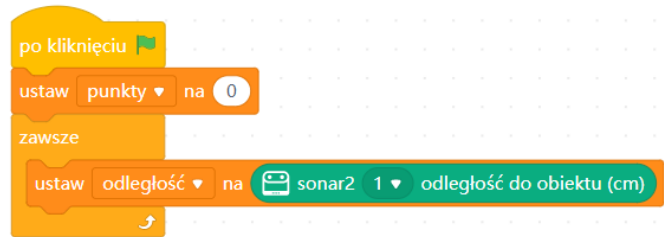
Skrypt dla niedźwiedzia:



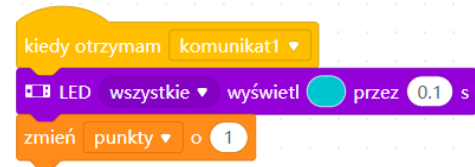
Skrypt dla miodu w wersji podstawowej:



Skrypt dla mBota:



Skrypt dla miodu w wersji rozszerzonej:



3. Podsumowanie i ewaluacja

Nauczyciel pyta uczniów, co najbardziej podobało się im w zajęciach, a co najmniej. Pyta także, czy tempo pracy nie było zbyt szybkie i czy jakieś kwestie zostały niewystarczająco wyjaśnione.

Uwagi/alternatywy:

Zachęcamy do utrwalania materiałów z przebiegu zajęć. Mogą to być zdjęcia, filmy, notatki, zrzuty ekranu, ciekawe lub zabawne teksty/informacje, które pojawiły się podczas lekcji. Posłużą one do dokumentowania przebiegu programu #SuperKoderzy.

Załącznik nr 1



- | | | |
|--------------|---------------------|--------------------------|
| 1. Turcja | 17. Serbia | 33. Białoruś |
| 2. Włochy | 18. Monako | 34. Liechtenstein |
| 3. Słowenia | 19. Luksemburg | 35. Bułgaria |
| 4. Estonia | 20. Chorwacja | 36. Bośnia i Hercegowina |
| 5. Litwa | 21. Finlandia | 37. Malta |
| 6. Belgia | 22. Kosowo | 38. Praga |
| 7. Szwecja | 23. Czarnogóra | 39. Holandia |
| 8. Dania | 24. Macedonia | 40. Niemcy |
| 9. Francja | 25. Hiszpania | 41. San Marino |
| 10. Łotwa | 26. Ukraina | 42. Watykan |
| 11. Albania | 27. Austria | 43. Rosja |
| 12. Andora | 28. Wielka Brytania | 44. Szwajcaria |
| 13. Węgry | 29. Polska | 45. Islandia |
| 14. Irlandia | 30. Norwegia | 46. Mołdawia |
| 15. Lizbona | 31. Słowacja | 47. Grecja |
| 16. Cypr | 32. Rumunia | |