MoboLab – roboty i tablety w Twojej szkole
Obszar I. „Zakoduj robota”

Scenariusze lekcji i zajęć pozalekcyjnych

SCENARIUSZ 7. AUTONOMICZNY POJAZD

scenariusz zajęć pozalekcyjnych

autor: Kamil Kociszewski
redakcja: Agnieszka Koszowska

SŁOWA KLUCZOWE:

mBot, mBlock, Arduino, autonomiczny

KRÓTKI OPIS ZAJĘĆ:

Podczas zajęć uczniowie i uczennice rozwijają wiedzę o środowisku programistycznym **mBlock** opartym na **języku Scratch** i konstruują autonomicznego robota. Poznają nową możliwość programowania robota i **tryb Arduino**, wgrywają program na płytkę Arduino. Programują robota do samodzielnego pokonania wyznaczonej trasy.

WIEDZA I UMIEJĘTNOŚCI ZDOBYTE PRZEZ UCZNIA / UCZENNICĘ:

- zna budowę robota mBot,
- zna podstawowe elementy interfejsu programu mBlock,
- swobodnie porusza się po środowisku mBlock, wie, gdzie szukać bloków do tworzenia skryptów sterujących czujnikami robota,
- potrafi stworzyć prosty program w środowisku programistycznym mBlock, wykorzystując czujnik linii, sygnały dźwiękowe i świetlne,
- zna projekt Arduino, wie jak zaprogramować robota mBot w trybie specjalnym Arduino,
- potrafi zaprogramować trasę, którą robot pokona samodzielnie.

GRUPA DOCELOWA:

Starsze klasy szkoły podstawowej (VII-) i klasy gimnazjalne (po dostosowaniu: możliwość realizacji w młodszych klasach: I-III i IV-VI szkoły podstawowej)

LICZBA UCZNIÓW / UCZENNIC W KLASIE:

Liczba optymalna: 12, liczba maksymalna: 16

CZAS TRWANIA ZAJĘĆ:

90 minut (lub 2 x 45 minut)

STOPIEŃ TRUDNOŚCI/SKOMPLIKOWANIA (w skali od 1 do 5 dla obszaru I. „Zakoduj robota”):

3

POTRZEBNY SPRZĘT I OPROGRAMOWANIE:

- komputer (przenośny lub stacjonarny),
- program mBlock (do pobrania ze strony: http://www.mblock.cc/download/),
- roboty mBot (złożone) – 1 robot na 1 ucznia / uczennicę, a w przypadku mniejszej liczby robotów: 1 robot na 2 lub 3 uczniów / uczennic,
- kable USB (po 1 dla każdego robota),
- projektor i laptop (w części teoretycznej).

CO NALEŻY PRZYGOTOWAĆ PRZED ZAJĘCIAMI:

- zainstalować program mBlock,
- sprawdzić poprawne działanie robota mBot oraz połączenie z programem mBlock (jeśli wystąpią problemy, warto zainstalować ponownie sterownik Arduino),
- dopasować stopień trudności zadania do potrzeb i możliwości klasy, dla której organizowane są zajęcia według wskazówek zawartych w scenariuszu,
- sprawdzić, czy wszystkie elementy robota są prawidłowo podpięte i czy brzęczyk oraz diody działają poprawnie,
- sprawdzić stan baterii zasilających robota.

KOMPETENCJE OSOBY PROWADZĄCEJ:

- zna i rozumie działanie wykorzystywanych bloków w programach Scratch i mBlock,
- potrafi podłączyć robota do komputera, używając kabla USB,
- wie, jakich bloków należy użyć do sterowania czujnikiem linii, brzęczykiem oraz do włączania i wyłączania diod,
- zna podstawowe pojęcia programistyczne (skrypt, program, pętla, instrukcja warunkowa),
- wie, dlaczego warto uczyć się programowania i jakie korzyści daje posiadanie umiejętności programistycznych,
- potrafi zachęcić do nauki programowania zarówno chłopców, jak i dziewczynki.

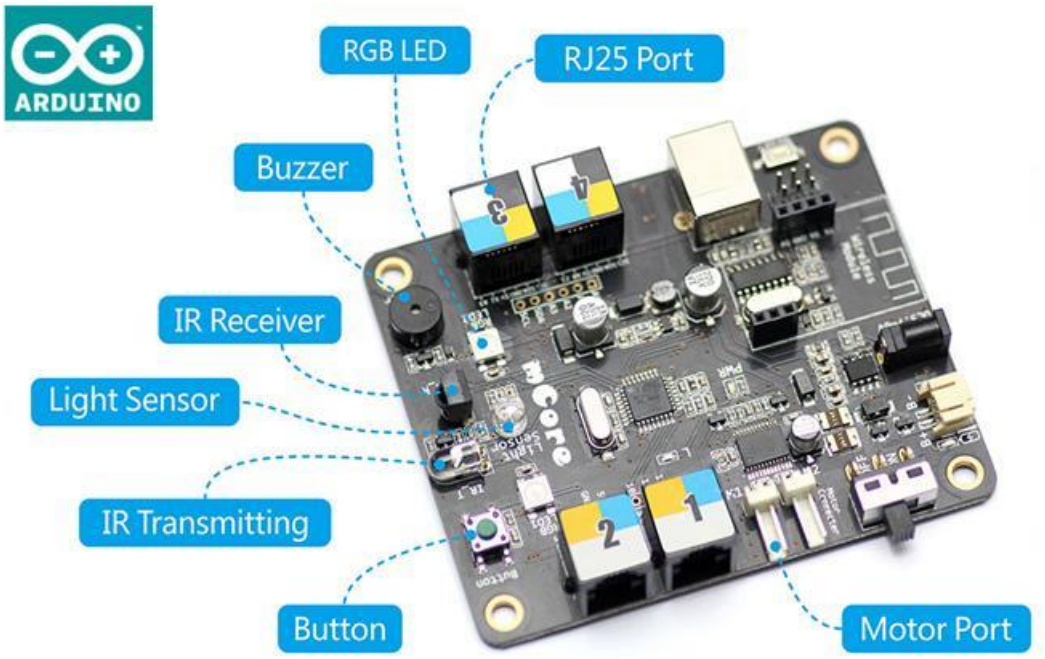
PRZEBIEG ZAJĘĆ:

Część 1 – 45 minut

Wprowadzenie – 5 minut

Cel: uczniowie i uczennice poznają Arduino – „mózg” robota mBot.
Zapowiadamy, że na tych zajęciach uczniowie i uczennice będą tak programować robota, by mógł on samodzielnie się poruszać. Pytamy uczniów, czy znają Arduino, a jeśli tak, to co wiedzą na temat tego urządzenia.

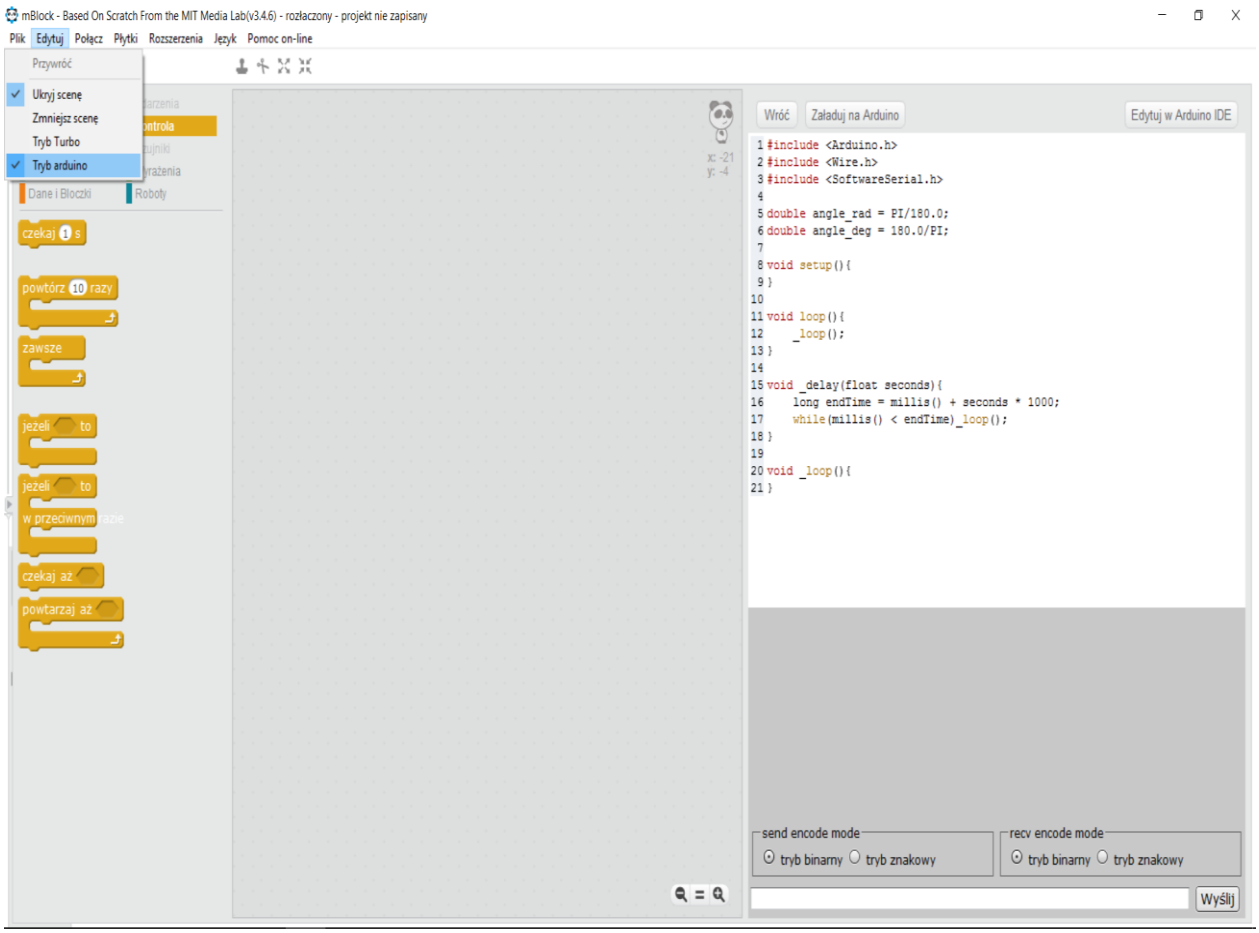
Arduino jest prostym mikrokontrolerem, który można programować i z jego pomocą samodzielnie konstruować różne urządzenia. Jest „mózgiem” takich urządzeń, jak drukarki 3D, roboty czy drony. Arduino znajduje się także w robocie **mBot**, lecz tu nosi nazwę **mCore**. Arduino jest „mózgiem” naszego robota, umożliwia „współpracę” z programem mBlock, a dodatkowo pozwala wykonywać samemu zadane wcześniej polecenia. Oprócz wejść, do których możemy podłączyć dodatkowe moduły (opisane między innymi w scenariuszu „Wprowadzenie do programu mBlock oraz podstawy sterowania robotem mBot”), Arduino ma także czujniki wbudowane na stałe w płytkę.



Omówienie trybu Arduino – 20 minut

Cel: uczniowie poznają tryb, który pozwala wgrywać programy na Arduino.

Wyjaśniamy, że do tworzenia programów, które będziemy mogli wgrać na płytkę Arduino w naszym robocie, wykorzystamy specjalny tryb dla Arduino. Umożliwia on stworzenie skryptu za pomocą specjalnych bloków, a następnie wgranie go na Arduino. Aby uruchomić tryb specjalny Arduino, należy z górnego paska programu mBlock wybrać polecenie „Edytuj” -> „Tryb Arduino”.



Zwracamy uwagę, że na ekranie zmienił się układ pola roboczego. Zamiast pola z „duszkiem”, w prawej części ekranu pojawił się nasz program „blokowy”, zapisany tym razem w języku zrozumiałym dla Arduino. Widoczny jest też przycisk „Załaduj na Arduino”, który umożliwia wgranie programu na Arduino. Ponadto większość kategorii duszków jest nieaktywna, ponieważ w tym trybie nie możemy z nich skorzystać.

Krótko przypominamy wykorzystywane bloki: Kontrola, Wyrażenia, Roboty. Krótko scharakteryzujemy te bloki:

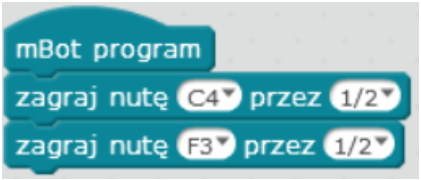
Roboty: bloki z tej kategorii służą do programowania interakcji z robotem – tworzenia skryptów, które umożliwiają sterowanie robotem i reakcję na zdarzenia oraz inicjowanie i kontrolę zdarzeń z udziałem poszczególnych elementów robota (np. czujników).

Kontrola: bloki z tej kategorii pozwalają sterować programem, na przykład dodawać do skryptu warunek, pętlę albo opóźniać wykonanie skryptu.
Wyrażenia: bloki z tej kategorii pozwalają wprowadzać do skryptu działania matematyczne lub wyrażenia logiczne.

Dodatkowo, w trybie Arduino aktywna jest kategoria **Dane i bloki**, której funkcje wykorzystamy na kolejnych zajęciach – dzięki nim, można tworzyć zmienne. Ważnym elementem tego trybu jest blok **mBot program** dostępny kategorii **Roboty**. Jest on początkiem każdego programu w tym trybie.

Istotne jest też to, że w trybie Arduino może powstać tylko **jeden program.** Poniższe rysunki pomogą zrozumieć, na czym polega różnica.

Prawidłowo stworzony program:



Nieprawidłowo stworzony program:

Jak widać na rysunkach, po wgraniu programu na Arduino można wykonać prawidłowo tylko jeden skrypt (jeden ciąg bloków).

Prosimy uczniów o przeciągnięcie kilku bloków i stworzenie dowolnego programu. Zwracamy uwagę, aby obserwowali zawartość **void setup()** w kodzie programu wgrywanego na płytkę. Dodanie nowych bloków zmieni zawartość tej części. Uczniowie mogą zaobserwować, w jaki sposób Arduino „rozumie” poszczególne bloki programu.

Pierwsze kroki z samodzielnym robotem – 20 minut

Cel: uczniowie tworzą pierwszy program umożliwiający samodzielne poruszanie się robota. Dajemy uczniom zadanie: mają stworzyć taki program, który pozwoli robotowi jechać do przodu, a następnie cofać się do punktu startowego. Uczniowie powinni wykorzystać wiedzę z poprzednich zajęć dotyczących ruchu robota. Proponujemy metodę prób i błędów: wgranie programu, sprawdzenie jego działania, poprawienie błędów, ponowne wgranie programu, sprawdzenie itd. Robot powinien przejechać około 20-50 cm, zatrzymać się, cofnąć w miejsce z którego ruszył i zatrzymać się na końcu trasy.

Przygotowane programy wgrywamy do robota, po uprzednim prawidłowym podłączeniu robota z komputerem, naciskając przycisk „Załaduj na Arduino”. Odległość, którą robot pokona jadąc do przodu lub do tyłu, modyfikujemy przy użyciu bloku opóźniającego „czekaj”, dostępnego w kategorii „Kontrola”.

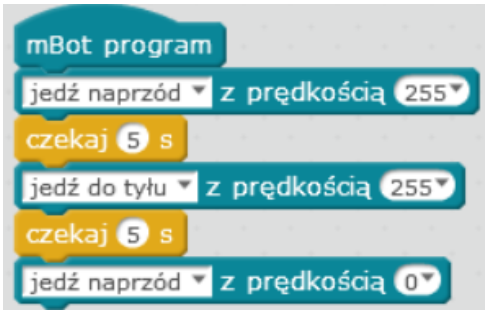
Uwaga: jeśli nie zastosujemy opóźnienia („czekaj”), ta część programu będzie wykonywana bardzo krótko, ponieważ jest to czas narzucony przez zegar Arduino. Natomiast, jeśli nie pojawi się kolejna instrukcja, to blok będzie wykonywał się bez przerwy. Dlatego na końcu skryptu dodajemy blok, który ustawi prędkość silników na

„0”.

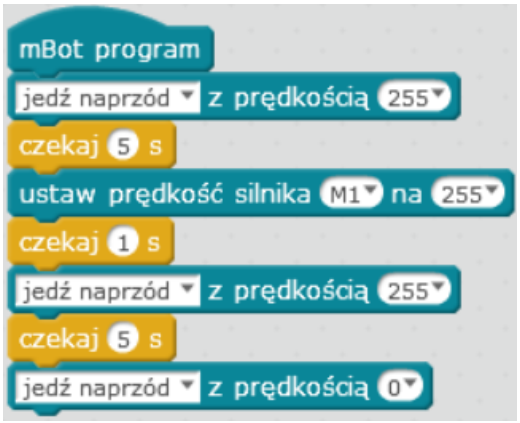
Zespołom, które zakończą zadanie wcześniej, zadajemy modyfikację programu: robot – zamiast cofania – powinien zawrócić (czyli obrócić się o 180 stopni).

Na poniższym rysunku widać przykładowy skrypt:

Uwaga: podane czasy czekania powinny być tak dobrane, aby robot prawidłowo wykonał zadanie (czasy widoczne na rysunku są przykładowe).



Zadanie dodatkowe:



W tym miejscu możliwa jest przerwa (kolejna część scenariusza będzie realizowana na następnej lekcji).

Część 2– 45 minut

Przypomnienie materiału z poprzedniej części zajęć – 10 minut

Rozpoczynamy od krótkiego przypomnienia materiału z poprzedniej części zajęć i odtworzenia powstałych skryptów.

Wspólne wyznaczenie trasy, którą robot będzie musiał pokonać – 5 minut

Cel: uczniowie wspólnie wyznaczają trasę, którą robot ma pokonać, a także ustalają

trudność tej trasy. Dajemy uczniom zadanie: zaplanowanie trasy, którą będzie musiał pokonać ich robot. Nie powinna ona być zbyt skomplikowana, może np. zawierać od 5 do 10 zakrętów w zależności od możliwości i poziomu grupy, bez przeszkód i na płaskiej podłodze. Punkty „start" i „meta" oraz zakręty można wyznaczyć za pomocą piórników, książek lub innych dostępnych przedmiotów.

Stworzenie programu, który pozwala robotowi pokonać trasę – 20 minut

Cel: uczniowie samodzielnie tworzą program sterujący robotem, dobierają parametry i poprawiają błędy, które zauważą podczas testów. Proponowany przebieg zadania:

Uczniowie rozpoczynają od ułożenia bloków sterujących silnikami, np. jazda naprzód, skręcaj w lewo, jazda naprzód… Następnie uzupełniają program o bloki „czekaj", pozwalające kontrolować odległość, którą pokona robot. Kolejnym krokiem jest sprawdzenie robota na wyznaczonej trasie. Ostatnim elementem jest wprowadzenie poprawek (np. wydłużenia lub skrócenia czasu czekania) i powtórzenie całego procesu po poprawkach. Jeśli grupy mają problem z pokonywaniem zakrętów, warto przedstawić im alternatywne rozwiązanie. Skręcanie w miejscu można realizować za pomocą bloku „Ustaw prędkość silnika …". Może to się przydać np. na trasach z ciasnymi zakrętami, gdzie lepiej jest tak zaprogramować robota, by obracał się w miejscu.

Zadanie zostanie wykonane, gdy robot pokona prawidłowo każdy zakręt i minie linię mety. Uczniów, którzy ukończą zadanie wcześniej, zachęcamy, by pracowali nad udoskonaleniem programu, tak aby robot jak najszybciej pokonywał trasę lub jego trasa była ciekawsza. Przykładowy program:

Pokonanie trasy na czas – 10 minut

Cel: sprawdzenie jak szybko roboty poszczególnych zespołów pokonują trasę.

Każdy z zespołów po kolei pozwala robotowi pokonać trasę mierząc czas. Uczniowie obserwują, jak poszczególne zespoły zaprogramowały robota. Członkowie zespołu, który uzyska najlepszy czas, krótko opowiadają pozostałym uczniom, w jak sposób udało się tego dokonać (np. poprzez wykorzystanie innych bloków lub parametrów). Uczestnicy zapisują program i wyłączają robota.

MOŻLIWE MODYFIKACJE DLA KLAS I-III I IV-VI:

W klasach I-III możliwe jest przeprowadzenie zajęć w formie zabawy. Uczniowie bawią się robotami, następnie pokazujemy im skrypty i wyjaśniamy, co oznaczają poszczególne bloki. Zachowujemy odpowiednio uproszczoną część teoretyczną, w części zadaniowej rozmawiamy z uczniami na temat sposobu wykonania zadań, realizujemy ich pomysły i tworząc program pokazujemy działanie robota.

W klasach IV-VI ułatwieniem może być wykorzystanie wcześniej przygotowanych szkieletów programów. Można wyjaśnić uczniom teorię, a następnie zadać proste zadanie (np. pracę z gotowym programem, do którego trzeba dobrać właściwe parametry, aby robot wykonał określone działanie).

ZADANIE SPRAWDZAJĄCE UMIEJĘTNOŚCI ZDOBYTE PODCZAS LEKCJI:

Uczeń / uczennica, pracując samodzielnie albo w dwu- lub trzyosobowym zespole tworzy skrypt sterujący robotem mBot. Za pomocą stworzonego skryptu steruje robotem tak, aby robot poruszał się samodzielnie na specjalnie zaprojektowanej trasie. Zespoły prezentują przejazdy swoich robotów na forum grupy, omawiając wykorzystane funkcje w programie.

PIGUŁKA WIEDZY I INSPIRACJI DLA OSÓB PROWADZĄCYCH:

Wykorzystywane kategorie bloków:

Zdarzenia: bloki z tej kategorii służą do programowania interakcji z użytkownikami – tworzenia skryptów, które reagują na określone działania użytkownika.

Roboty: bloki z tej kategorii służą do programowania interakcji z robotem – tworzenia skryptów, które umożliwiają sterowanie robotem i reakcję na zdarzenia oraz inicjowanie i kontrolę zdarzeń z udziałem poszczególnych elementów robota (np. czujników).

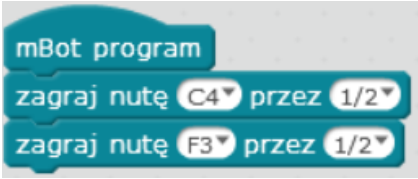
Kontrola: bloki z tej kategorii pozwalają sterować programem, na przykład dodawać do skryptu warunek, pętlę albo opóźniać wykonanie skryptu.

Wyrażenia: bloki z tej kategorii pozwalają wprowadzać do skryptu działania matematyczne lub wyrażenia logiczne.

Tryb Arduino pozwala stworzyć program z wykorzystaniem specjalnych bloków, a następnie wgrać go na Arduino. Aby uruchomić tryb Arduino, należy z górnego paska opcji wybrać „Edytuj” -> „Tryb Arduino”.

W trybie Arduino może powstać tylko **jeden program.**

Prawidłowo stworzony program:



Nieprawidłowo stworzony program:

Scenariusz został opracowany na potrzeby projektu „MoboLab – roboty i tablety w Twojej szkole”. Celem projektu jest zwiększenie kompetencji informatycznych z zakresu programowania i wykorzystywania technologii mobilnych w uczeniu się, a także kreatywności, innowacyjności i umiejętności współpracy w zespole z wykorzystaniem TIK, uczniów / uczennic z (UCZ) z 6 szkół podnadgimnazjalnych i 4 gimnazjów Wołomina i Zielonki. Projekt dofinansowany jest ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego (Regionalny Program Operacyjny Województwa Mazowieckiego na lata 2014-2020, Oś Priorytetowa X. Edukacja dla rozwoju regionu, Działanie 10.1. Edukacja ogólna i przedszkolna, Poddziałanie 10.1.2. Edukacja ogólna w ramach ZIT).