

MoboLab – roboty i tablety w Twojej szkole

Obszar I. „Zakoduj robota”

Scenariusze lekcji i zajęć pozalekcyjnych

SCENARIUSZ 12. POJAZD PODĄŻAJĄCY ZA ŚWIATŁEM

scenariusz zajęć pozalekcyjnych

autor: Kamil Kociszewski

redakcja: Agnieszka Koszowska

SŁOWA KLUCZOWE:

mBot, mBlock, Arduino, autonomiczny, czujnik natężenia światła, zmienne, pilot

KRÓTKI OPIS ZAJĘĆ:

Podczas zajęć uczniowie i uczennice rozwijają wiedzę o środowisku programistycznym **mBlock** opartym na języku **Scratch**. Za pomocą bloków służących do programowania **czujnika światła** oraz **pilota**, a także wykorzystując **zmienne** konstruują robota podążającego za światłem. Zajęcia można przeprowadzić w formie eksperymentu lub zabawy.

WIEDZA I UMIEJĘTNOŚCI ZDOBYTE PRZEZ UCZNIĄ / UCZENNICĘ:

- zna budowę robota mBot,
- zna podstawowe elementy interfejsu programu mBlock,
- wie, co to jest czujnik światła,
- swobodnie porusza się po środowisku mBlock, wie, gdzie szukać bloków do tworzenia skryptów sterujących robotem,
- potrafi stworzyć skrypt sterujący robotem przy pomocy czujnika światła,
- wie, co to jest „tryb Arduino” i potrafi z niego skorzystać,
- potrafi wykorzystać zmienne aby sterować programem,
- potrafi sterować programem za pomocą pilota.

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 min (lub 2 x 45 minu)

STOPIEŃ TRUDNOŚCI/SKOMPLIKOWANIA

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

1 w wersji mniej zaawansowanej, 4/5 w wersji zaawansowanej

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),
- latarka lub telefon z latarką,
- karton i taśma klejąca,
- sala, w której można zapewnić zaciemnione miejsce (np. zasłonić okna).

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 diody działają poprawnie,
- sprawdzić stan baterii zasilających robota,
- przed tymi zajęciami bardzo ważne jest dokładne zapoznanie się ze scenariuszem i wybranie metody realizacji zajęć, warto też przypomnieć sobie poprzednie scenariusze lub stworzone wcześniej programy dotyczące poruszanych zagadnień.

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 robotem,
- zna podstawowe pojęcia programistyczne (skrypt, program, pętla),

- 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

Uwaga! Zajęcia można przeprowadzić na dwa sposoby, w zależności od poziomu zaawansowania grupy. Pierwszy sposób (dla mniej zaawansowanych grup) to stworzenie programu dla robota podążającego za światłem, a następnie zabawa z robotem i ulepszenia w programie.

Drugi sposób (dla grup bardziej zaawansowanych) to stworzenie programu dla robota podążającego za światłem, a dodatkowo dołączenie sterowania za pomocą pilota. W scenariuszu opisano drugi sposób realizacji zajęć. W przypadku wyboru pierwszego sposobu można pominąć sterowanie pilotem, a na drugiej części zajęć przeprowadzić gry i zabawy z zaprogramowanymi robotami.

Wprowadzenie – 5 minut

Przedstawiamy temat zajęć. Informujemy, że na tych i następnych zajęciach uczniowie będą programować robota, który podąża za światłem. Robot zostanie zaprogramowany za pomocą „trybu Arduino” i będzie poruszał się samodzielnie.

Omówienie sposobu działania robota – 10 minut

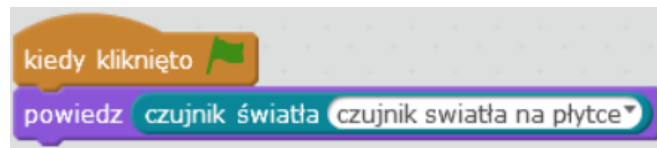
Pytamy uczniów, czy mają jakiś pomysł na to, jak taki robot mógłby działać. Robot powinien szukać światła i podążać w jego kierunku. Autor scenariusza proponuje następujące rozwiązanie: robot kręci się w kółko, a gdy „zobaczy” światło, jedzie do przodu w jego kierunku. Jeśli pojawią się inne pomysły i będą wykonalne, można wypróbować różne sposoby realizacji zadania.

Przypominamy położenie czujnika światła na płytce robota. Pytamy uczniów, czy mają jakieś obawy związane z szukaniem kierunku, z którego pada światło. Chodzi o to, że aby móc zrealizować zadanie, powinniśmy wprowadzić do naszego robota pewne modyfikacje za pomocą kartonu i taśmy. Problemem który trzeba rozwiązać, jest umiejscowienie czujnika. Jest on skierowany do góry i dlatego może nie być w stanie prawidłowo rozpoznać kierunku, z którego pada światło. Rozwiązaniem jest zabudowanie górnej części robota, tak aby z czterech stron czujnik był zasłonięty, a z jednej strony (przód) odkryty – podobnie jak domek z kartonu bez przedniej ściany. Wystarczy przykleić po kawałku kartonu z każdej ze stron: góra, tył, lewa strona, prawa strona, i w ten sposób zmodyfikować robota.

Kalibracja czujnika – 10 minut

Przypominamy, że aby przystąpić do prac nad programem, najpierw należy przeprowadzić tzw. kalibrację czujnika światła. Robot podążający za światłem będzie działać w ciemnym miejscu, trzeba więc dokonać wstępnego pomiaru natężenia światła, aby móc stworzyć prawidłowy skrypt.

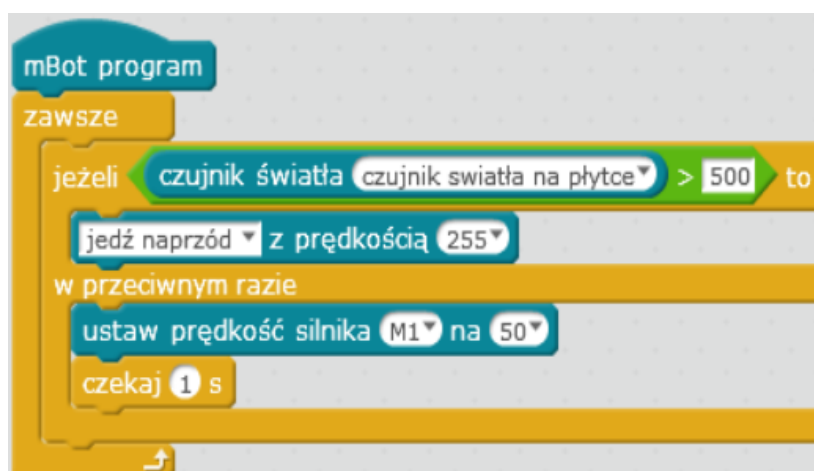
Uczniowie tworzą program umożliwiający przeprowadzenie kalibracji, podłączają robota do komputera i zapisują odczytaną wartość. Wartość powinno się sprawdzić w sytuacji, kiedy czujnik robota jest zaciemniony. Takie zaciemnienie możemy zasymulować np. zasłaniając ręką jedyną odkrytą stronę czujnika (czyli przód kartonowego „domku”). Jeśli uczniowie nie pamiętają, w jaki sposób można sprawdzić tę wartość, pokazujemy im program stworzony na zajęciach dotyczących czujnika.



Zapisaną wartość wykorzystamy w kolejnych etapach pracy nad programem. Może się zdarzyć, że wartość została dobrana nieprawidłowo i będzie trzeba ją poprawić po testach w zaciemnionym pomieszczeniu.

Tworzenie programu umożliwiającego robotowi szukanie światła – 20 minut

Uczniowie samodzielnie przygotowują skrypt dla tego zadania. Jeśli na zajęciach jest realizowana propozycja autora scenariusza, to robot ma się kręcić w kółko, a po wykryciu światła – jechać do przodu w jego kierunku. Program może wyglądać tak, jak na poniższym rysunku:



W tym miejscu możliwa jest przerwa (kolejna część scenariusza będzie realizowana na następnych zajęciach).

Cz. 2 – 45 minut

Uwaga! Przebieg drugiej części zajęć jest uzależniony od poziomu zaawansowania grupy. W przypadku grup mniej zaawansowanych drugą część można przeprowadzić w postaci gier i zabaw z robotami, z uwzględnieniem powtórek czy ewentualnych zmian w kalibracji czujnika.

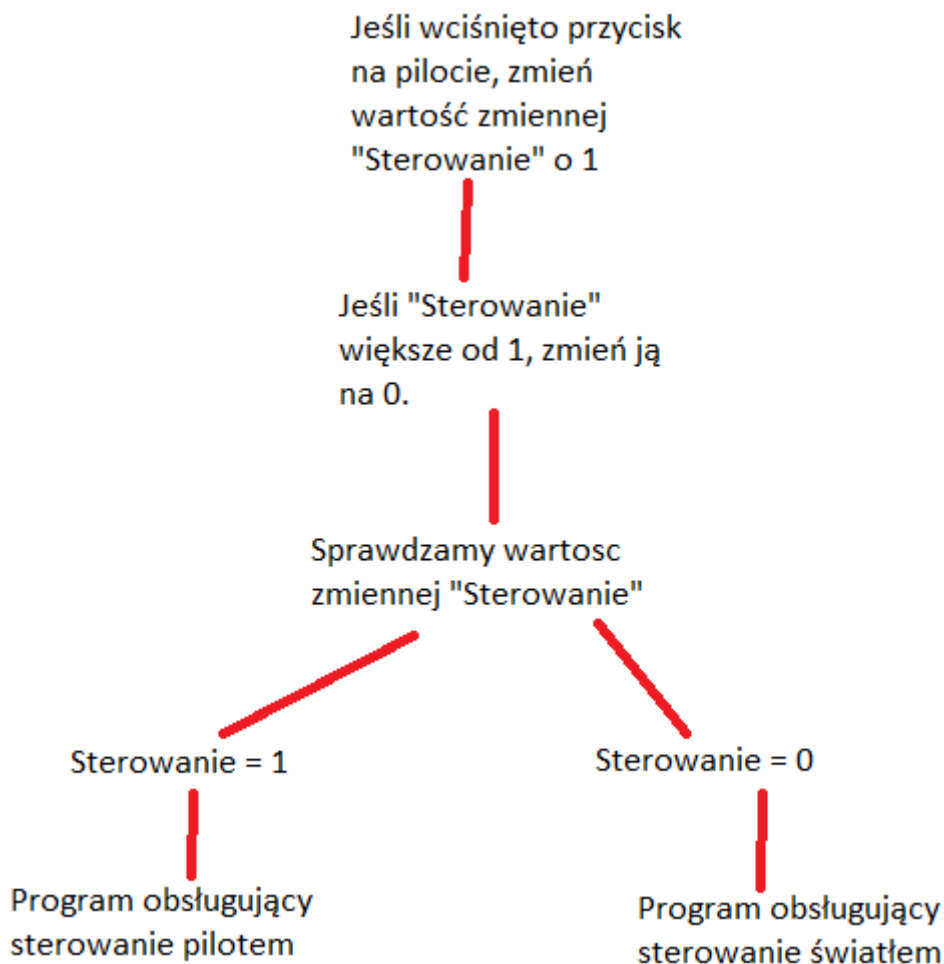
W przypadku grup bardziej zaawansowanych, druga część zajęć będzie polegać na dodaniu nowych funkcji robota.

Omówienie zadania do wykonania w drugiej części zajęć – 10 minut

Uwaga! W przypadku wyboru tego sposobu realizacji zajęć na sprawdzenie działania zaprogramowanego robota może nie wystarczyć czasu – zwłaszcza takim zespołom, które pracują wolniej. Można pozostawić przetestowanie programów na następne zajęcia albo spróbować zakończyć zadanie wcześniej (np. w razie potrzeby przyspieszyć proces tworzenia programów udostępniając uczniom podpowiedzi lub „puste” skrypty do uzupełnienia).

Omawiamy kolejne zadanie do wykonania: dodanie awaryjnego sterowania robotem za pomocą pilota. Będzie to alternatywny sposób sterowania, dzięki któremu uczniowie będą mogli dodać do programu nowe funkcje, a także powtórzyć materiał z poprzednich zajęć dotyczących wykorzystywania pilota i zmiennych.

Najpierw odtwarzamy część programu z poprzedniej części zajęć, dotyczącej sterowaniem światłem. Następnie rysujemy na tablicy algorytm sterowania.



Jeśli wybrano sterowanie pilotem, robot reaguje na polecenia pilota. Jeśli wybrano sterowanie światłem, robot poszukuje źródła światła.

Wybór sterowania ma się odbywać poprzez naciśnięcie przycisku na pilocie. Najprościej jest przedstawić to w programie za pomocą zmiennych. Jeśli naciśniemy przycisk, to zmienna „Sterowanie” zmieni się z zera na jedynkę, co oznacza wybranie sterowania pilotem. Jeśli kolejny raz naciśniemy przycisk, to zmienna zmieni się o kolejny jeden. W naszym przypadku przyjmujemy, że sterowanie światłem realizuje się dla wartości zmiennej „Sterowanie” równej 0.

Teraz przedstawiamy bardzo ważny problem programistyczny, który może się przydać również w innych sytuacjach. Program działa w taki sposób, że kolejne naciśnięcie przycisku na pilocie spowoduje zmianę wartości zmiennej na 2. Natomiast my chcemy uzyskać wartość 0. Aby to zrealizować, wystarczy przy zmianie sprawdzić, czy wartość jest większa od 1 i jeśli tak, to zmieniamy ją na 0. W ten sposób będziemy otrzymywać zawsze wartość 0 lub 1.

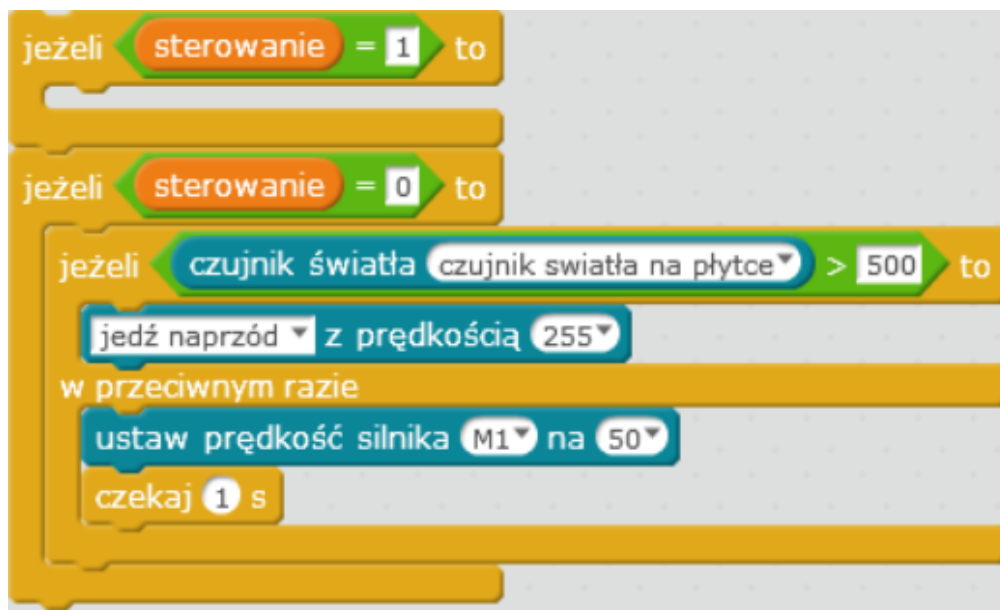
Dodatkowo, można dodać jeszcze jedną zmienną „Światło”, która pozwoli zapalać i gasić światła robota przez naciśnięcie przycisku.

Realizacja zadania – 25 minut

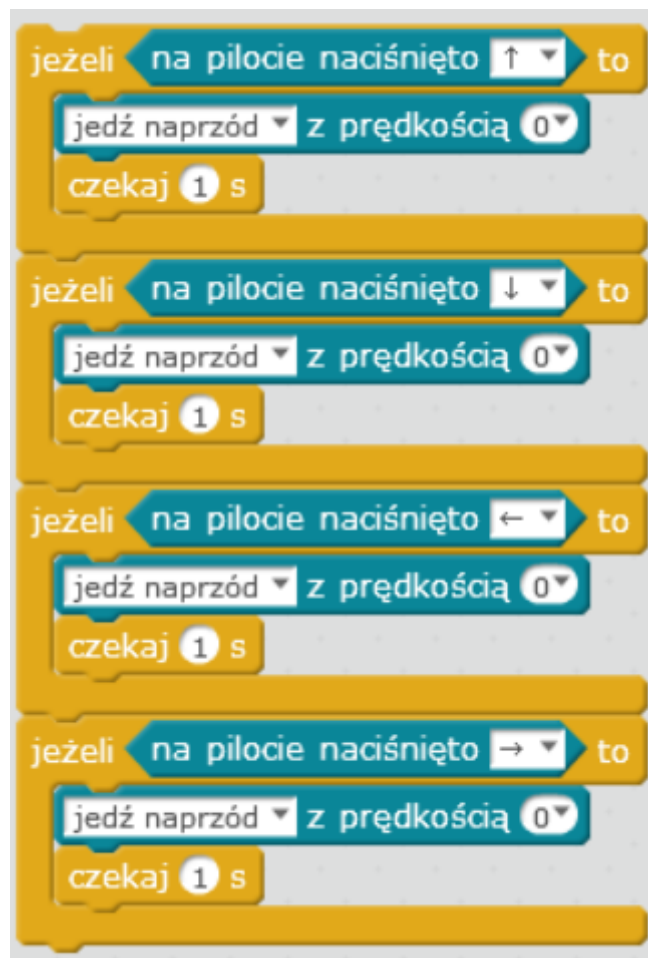
Uczniowie wykonują zadanie. Jako ewentualną wskazówkę, można im zaproponować kolejność prac do wykonania: najpierw należy stworzyć zmienne i „obsługę” ich zmiany za pomocą pilota.



Następnie tworzymy dwie instrukcje warunkowe, sprawdzające wartość zmiennej.



Następnie wypełniamy funkcję warunkową dotyczącą sterowania pilotem. Wykorzystujemy do tego wiedzę z poprzednich zajęć.



Prezentacja końcowego programu – 10 minut

Prezentujemy cały program. Wydaje się on skomplikowany, lecz po prostu składa się z wielu powtarzających się bloków. Uczniowie wprowadzają ewentualne poprawki, a następnie wgrywają program do robota za pomocą polecenia „Załaduj na Arduino” i koniecznie **ZAPISUJĄ PROGRAM** w celu naniesienia poprawek w kalibracji po testach. Testy powinny się przeprowadzić w ciemnym miejscu z wykorzystaniem latarki.


```
mBot program
zawsze
  jeżeli na pilocie naciśnięto A to
    zmień sterowanie o 1
    jeżeli sterowanie > 1 to
      ustaw sterowanie na 0
  jeżeli na pilocie naciśnięto B to
    zmień światło o 1
    jeżeli światło > 1 to
      ustaw światło na 0
  jeżeli sterowanie = 1 to
    jeżeli na pilocie naciśnięto ↑ to
      jedź naprzód z prędkością 0
      czekaj 1 s
    jeżeli na pilocie naciśnięto ↓ to
      jedź naprzód z prędkością 0
      czekaj 1 s
    jeżeli na pilocie naciśnięto ← to
      jedź naprzód z prędkością 0
      czekaj 1 s
    jeżeli na pilocie naciśnięto → to
      jedź naprzód z prędkością 0
      czekaj 1 s
```

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 i jego czujnikiem światła. Za pomocą stworzonego skryptu steruje robotem tak, aby robot podążał za światłem.

FIGUŁKA 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).

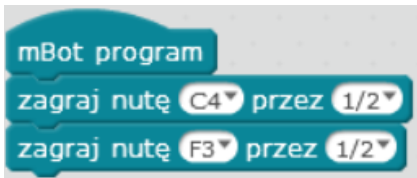
Kontrola: bloki z tej kategorii pozwalają sterować programem, na przykład dodawać do skryptu warunek, pętlę albo opóźnić wykonanie skryptu.

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

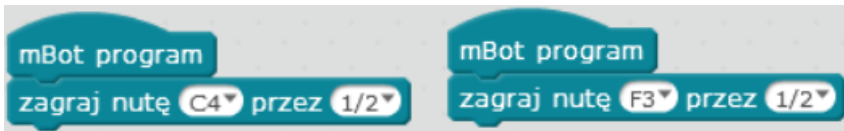
Dane i bloki: bloki z tej kategorii pozwalają na tworzenie i obsługę zmiennych.

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”.

Prawidłowo stworzony program:

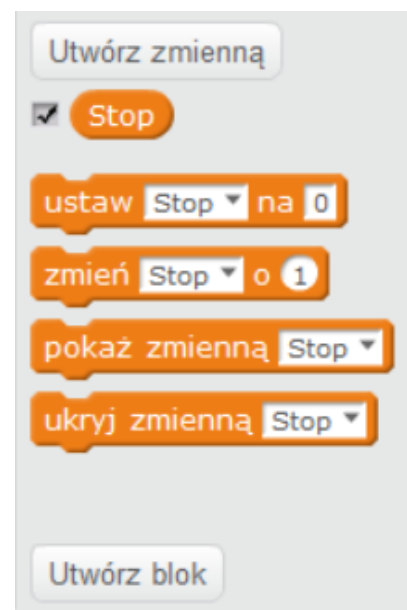
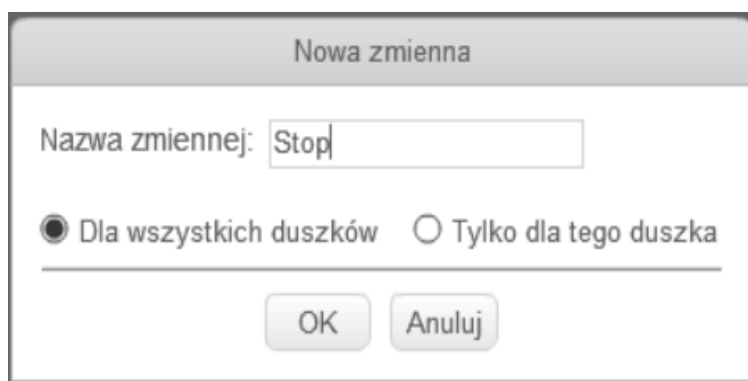


Nieprawidłowo stworzony program:



Jak widać, po wgraniu na płytkę Arduino prawidłowo wykona **tylko jeden ciąg bloków**.

Zmienne znajdziemy w kategorii „Dane i bloczki”. Po kliknięciu w tę kategorię, zobaczymy dwa polecenia: „Utwórz zmienną” oraz „Utwórz blok”. Na naszych zajęciach będziemy wykorzystywać tylko pierwsze z nich. Gdy klikniemy w „Utwórz zmienną”, wyświetli się okno, w którym możemy nadać nazwę zmiennej oraz zdecydować, dla jakich „duszków” ta zmienna będzie dostępna. W programach tworzonych na tych zajęciach skorzystamy z opcji „Dla wszystkich duszków”.



Po utworzeniu zmiennej, pojawiają się kolejne polecenia w kategorii „Dane i bloczki”.

Pierwszy blok „ustaw... na...” pozwala ustawić wartość naszej zmiennej. Kolejny blok – „zmień... o...” pozwala zmienić wartość zmiennej o daną liczbę. Bloki „pokaż zmienną...” i „ukryj zmienną...” umożliwiają wyświetlenie wartości naszej zmiennej oraz ukrycie jej podczas pracy z duszkiem.

Program możemy uruchomić ponownie przy pomocy przycisku RESET znajdującego się na płytce robota.

Program podążający za światłem:



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).



Ten utwór jest dostępny na licencji [Creative Commons Uznanie autorstwa 4.0 Międzynarodowe](https://creativecommons.org/licenses/by/4.0/).