

MoboLab – roboty i tablety w Twojej szkole

Obszar I. „Zakoduj robota”

Scenariusze lekcji i zajęć pozalekcyjnych

SCENARIUSZ 14. INTERAKCJE Z DUSZKAMI

scenariusz zajęć pozalekcyjnych

autor: Michał Kłosiński

redakcja: Agnieszka Koszowska

SŁOWA KLUCZOWE:

mBlock, mBot, Scratch, robot, sterowanie

KRÓTKI OPIS ZAJĘĆ:

Podczas zajęć uczniowie i uczennice rozwijają wiedzę o środowisku programistycznym **mBlock** opartym na języku **Scratch**. Poznają bloki służące do budowania skryptów, za pomocą których można zaprogramować robota **mBot** oraz „duszki” stanowiące element interfejsu **Scratcha** w taki sposób, aby robot wpływał na parametry duszka.

WIEDZA I UMIEJĘTNOŚCI ZDOBYTE PRZEZ UCZNIĄ / UCZENNICĘ:

- zna budowę robota mBot, zna zastosowanie czujnika odległości robota,
- zna podstawowe elementy interfejsu programu mBlock opartego na środowisku Scratch,
- swobodnie porusza się po środowisku mBlock, wie, gdzie szukać bloków do tworzenia skryptów sterujących robotem,
- potrafi stworzyć prosty skrypt sterujący czujnikiem odległości robota,
- potrafi stworzyć program w mBlock, który pozwoli na interakcję duszka z robotem poprzez odczyt czujników robota.

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 45min)

STOPIEŃ TRUDNOŚCI/SKOMPLIKOWANIA

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

1 (3 wraz z zadaniami dodatkowymi)

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 poprzez Bluetooth,
- przed przystąpieniem do połączenia warto wgrać program domyślny używając kabla USB: polecenie „Przywróć program domyślny” z menu „Połącz”; w programie domyślnym jest program obsługujący Bluetooth, bez niego mogą wystąpić problemy z połączeniem,
- UWAGA! interakcja robota z duszkami jest możliwa jedynie poprzez wykorzystanie połączenia Bluetooth,
- dopasować stopień trudności zadania do potrzeb i możliwości klasy, dla której organizowane są zajęcia według wskazówek zawartych w scenariuszu,
- sprawdzić stan baterii zasilających robota.

KOMPETENCJE OSOBY PROWADZĄCEJ:

- zna i rozumie działanie wykorzystywanych bloków w programach Scratch i mBlock,
- potrafi podłączyć robota do komputera poprzez Bluetooth,
- wie, jakich bloków należy użyć do sterowania ruchem robota, włączania i wyłączania diod,
- 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

Wprowadzenie – 10 minut

Wprowadzamy uczniów w tematykę zajęć. Zapowiadamy, że na dzisiejszych zajęciach podłączymy robota bezprzewodowo poprzez Bluetooth i będziemy nim sterować za pomocą programu napisanego w mBlock. Włączenie przycisku „Zielona flaga” spowoduje uruchomienie naszego programu. Program nie zostanie „przekopiowany”, czyli wgrany do robota, ale robot dzięki połączeniu bezprzewodowemu będzie reagował na polecenia naszego programu.

Program mBlock jest zgodny z programem Scratch 2.0. Oznacza to, że zawiera on wszystkie polecenia (i elementy interfejsu) Scratcha i posiada dodatkowe polecenia obsługujące mBota oraz jego czujniki. Scratch jest programem dostępnym zarówno w wersji stacjonarnej (do instalacji na komputerze), jak i w wersji online poprzez stronę www.scratch.mit.edu.

W wersji online możemy korzystać z jego wszystkich funkcjonalności, jeśli utworzymy w nim konto i się zarejestrujemy. Do rejestracji potrzebny jest adres poczty elektronicznej. W przypadku gdy nie założymy konta, możemy tworzyć programy, lecz nie mamy możliwości ich zapisania. Na stronie www.scratch.mit.edu, w dziale „Pomoc” znajdziemy materiały pomocne również w tworzeniu programów w mBlock.

Korzystając z mBlock i tworząc własny program, możemy wpływać na zachowania robota, ale działa to również w drugą stronę: robot i jego czujniki mogą mieć wpływ na działanie duszków w naszym programie. Tym właśnie będziemy się zajmować na tych i następnych zajęciach.

Łączenie robota z programem przez Bluetooth

Przed przystąpieniem do połączenia warto wgrać program domyślny używając kabla USB: korzystamy z polecenia „Przywróć program domyślny” (mBot) z menu „Połącz”. W programie domyślnym znajduje się program obsługujący moduł Bluetooth. Aby połączyć robota z programem należy:

- ⇒ włączyć robota,
- ⇒ z menu programu mBlock wybrać polecenie „Połącz”, a następnie Bluetooth i wreszcie polecenie „Wykryj”.

Po chwili pojawi się okno z listą dostępnych urządzeń do połączenia. Uwaga: podczas pracy z wieloma robotami może nastąpić zamieszanie z wyborem właściwego robota.

Roboty można opisać przed zajęciami – każdy z nich posiada unikalny, dwunastocyfrowy numer MAC. Innym rozwiązaniem może być łączenie robotów w każdym zespole po kolei (tylko jeden robot jest włączony – pozostałe są wyłączone).

Po wybraniu urządzenia z listy system poprosi o „sparowanie” urządzenia poprzez potwierdzenie numeru. Numer nie jest wyświetlany na robocie, ale potwierdzamy go. Po prawidłowym sparowaniu nastąpi instalacja sterowników. O tym, że połączenie zakończyło się sukcesem, powiadomi nas okienko dialogowe „Bluetooth połączony”.

W programie mBlock na liście dostępnych połączeń powinno być dostępne połączone urządzenie – nasz robot. Program mBlock zapamiętuje połączone urządzenia, dzięki temu przy kolejnym uruchomieniu programu (lub robota) będzie on już dostępny na liście urządzeń.

Określenie warunków programu spełniających założone zadanie – 5 min

Cel: określenie warunków, które musi posiadać program, aby spełniał kryteria sukcesu. Należy zastanowić się:

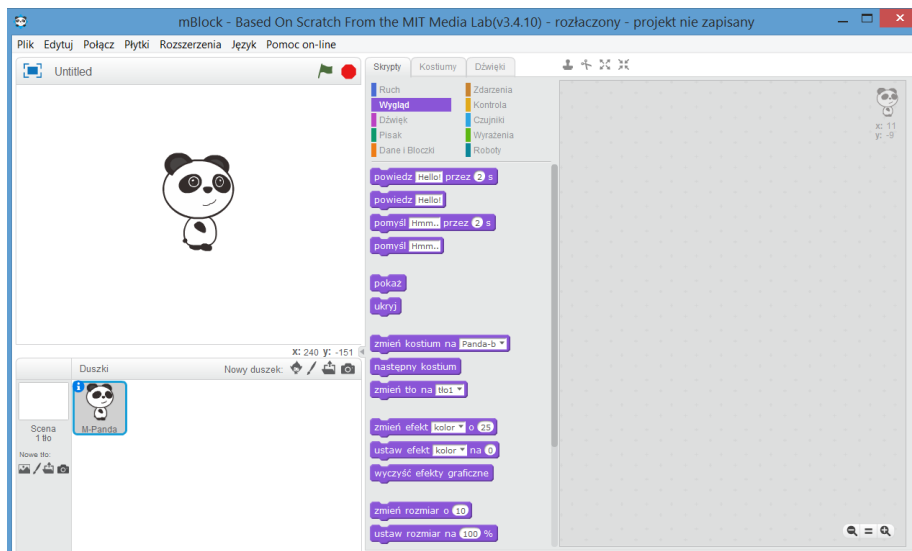
- ⇒ *Jak będzie działał nasz program?*
- ⇒ *W jaki sposób robot będzie oddziaływał na duszka?*

Program powinien działać następująco: jeśli czujnik odległości robota wykryje przeszkodę w promieniu 50 cm, duszek zareaguje i powie „Uwaga! Przeszkoda!”.

Dzielimy uczniów na zespoły. Liczba osób w zespole zależy od liczby robotów, które mamy do dyspozycji (optymalnie: 1 robot na 2 osoby + 1 robot dla osoby prowadzącej lekcję). Każdy zespół otrzymuje jednego robota.

Budowa programu – 30 min

Cel: zaprogramowanie duszka, który po wykryciu przez robota przeszkody w promieniu 50 cm powie „Uwaga! Przeszkoda!”. Potrzebny będzie duszek – „standardowym” duszkiem w programie mBlock jest panda.



Jeśli chcemy zmienić duszka, można wykorzystać dostępną w mBlock bibliotekę duszków, narysować własną postać albo wczytać plik z dysku lub z kamery.



Dla wybranego duszka układamy bloki:



Testujemy działanie programu poprzez jego uruchomienie – kliknięcie w zieloną flagę.



Następnie rozbudowujemy nasz program. W zależności od odległości od przeszkody duszek ma „powiedzieć” coś innego, np. w przypadku odległości od 0 do 10 cm będzie

to: „Jesteś zbyt blisko!”, między 10 a 30 cm: „Jesteś blisko przeszkody”, a między 30 a 50cm: „Przeszkoda przed nami”. Potrzebne będą bloki:



Układamy warunek dotyczący przedziału odległości, korzystając z bloku sterującego czujnikiem:



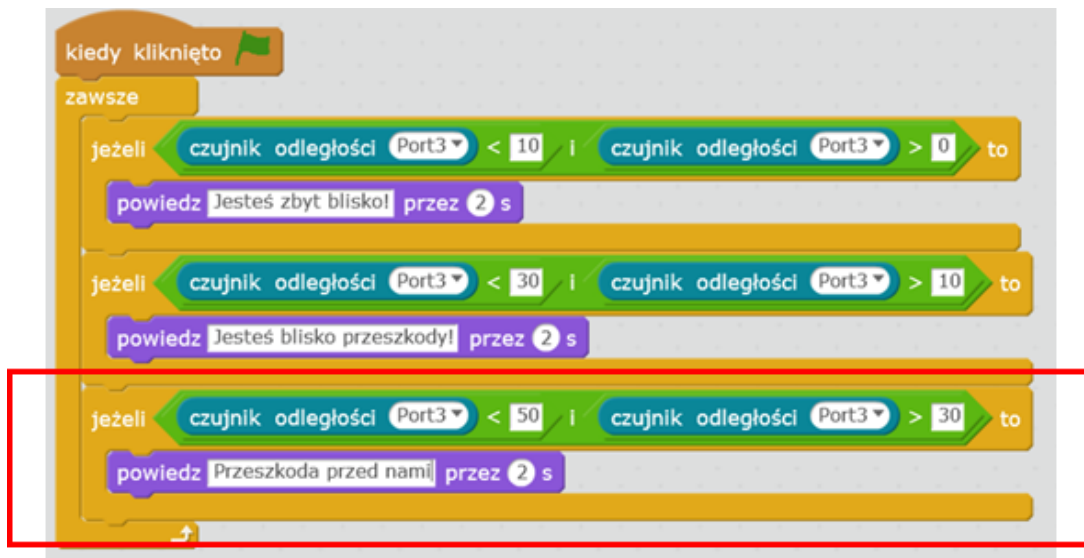
Cały program (z warunkiem dotyczących odległości między 0 a 10 cm) będzie wyglądał następująco:



Teraz dokładamy kolejny przedział odległości:



I kolejny:



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

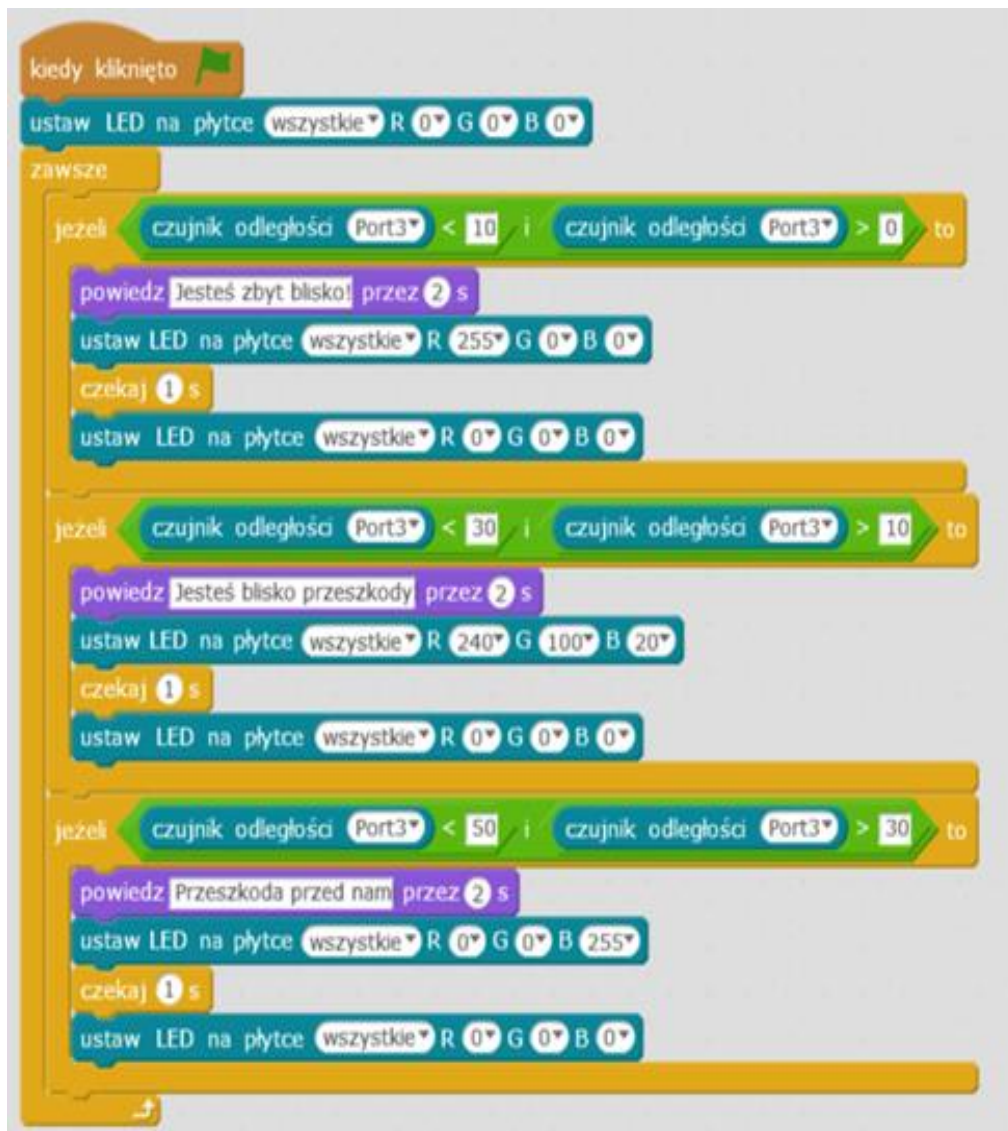
Część 2. – 45 minut

Przypomnienie materiału z poprzedniej części zajęć – 15 minut

Rozpoczynamy od krótkiego przypomnienia materiału z poprzedniej części zajęć i odtworzenia skryptów, służących do sterowania robotem oraz programowania interakcji między duszkiem i robotem.

Modyfikacje programu – 20 minut

Oprócz komunikatów duszka wyświetlanych na ekranie, mogą pojawiać się też inne sygnały informujące o odległości robota od przeszkody, np. zapalające się światła diod o różnych kolorach. (Temat ustawienia kolorów innych niż czerwony, zielony, niebieski – RGB został omówiony w scenariuszu nr 4. *Poznajemy i programujemy czujnik odległości i brzęczyk*). Modyfikujemy programy w taki sposób, by uwzględnić zapalanie świateł robota o określonych kolorach. Program może wyglądać tak jak na poniższym rysunku:



Możemy też dodać – oprócz świateł – sygnał dźwiękowy. A więc, robot jedzie do przodu, po wykryciu przeszkody, w zależności od odległości od niej, duży komunikat informuje o zbliżeniu się do przeszkody, a robot sygnałami świetlnymi. Przy odległości 5 cm od przeszkody robot zatrzymuje się i wydaje dźwięk. Program może wyglądać tak, jak na poniższym rysunku:



Podsumowanie – 10 min

Krótko podsumowujemy materiał z zajęć. Uczniowie zapisują program, wyłączają robota i porządkują stanowiska pracy.

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).

W młodszych klasach można przeprowadzić zajęcia w uproszczonej formie, np. bez warunków szczegółowo określających przedziały odległości. Można ograniczyć warunki do dwóch przedziałów wartości odległości (lub każdy zespół otrzymuje inny przedział

odległości i wymienia się doświadczeniami). Należy więcej czasu przeznaczyć na testowanie programów.

ZADANIE SPRAWDZAJĄCE UMIEJĘTNOŚCI ZDOBYTE PODCZAS LEKCJI:

Uczeń / uczennica, pracując samodzielnie albo w dwu- lub trzyosobowym zespole tworzy skrypt sterujący robotem mBot. Za pomocą skryptu steruje ruchem robota, czujnikiem odległości robota oraz duszkiem w programie mBlock – w taki sposób, by na ekranie komputera wyświetlały się komunikaty duszka informujące o zbliżaniu się robota do przeszkody. Komunikaty mają różną treść, w zależności od odległości robota od przeszkody.

PIGUŁKA WIEDZY I INSPIRACJI DLA OSÓB PROWADZĄCYCH:

Informacje dotyczące połączenia robota przez Bluetooth w różnych systemach operacyjnych można znaleźć na stronie:

<http://www.trobot.pl/materialy-dydaktyczne/jak-zbudowac-robota-pobierz-instrukcje>

Materiały, inspiracje, do zajęć i przykładowe skrypty można znaleźć m. in. w scenariuszach lekcji i zajęć z podstaw programowania w środowisku Scratch dostępnych na stronie:

http://wiki.mistrzowiekodowania.pl/index.php?title=Strona_g%C5%82%C3%B3wna

Oficjalna strona scratch dostępna jest pod adresem:

<https://scratch.mit.edu/>

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/).