

MoboLab – roboty i tablety w Twojej szkole

Obszar I. „Zakoduj robota”

Scenariusze lekcji i zajęć pozalekcyjnych

SCENARIUSZ 6. ŁĄCZYMY CZUJNIKI

scenariusz zajęć pozalekcyjnych

autor: Kamil Kociszewski

redakcja: Agnieszka Koszowska

SŁOWA KLUCZOWE:

mBot, mBlock, Arduino, sterowanie, diody, czujnik odległości, czujnik linii, czujniki

KRÓTKI OPIS ZAJĘĆ:

Podczas zajęć uczniowie i uczennice budują **interaktywnego robota** programując w języku **mBlock** różne **czujniki** robota **mBot**. Wykorzystując wiedzę z wcześniejszych lekcji lub zajęć dotyczących programowania tworzą proste skrypty służące do sterowania robotem mBot i rozszerzają je o nowe funkcje. Na zakończenie wykonują zadanie: uruchamiają robota, który porusza się i reaguje na otoczenie.

WIEDZA I UMIEJĘTNOŚCI ZDOBYTE PRZEZ UCZNIĄ / UCZENNICĘ:

- swobodnie porusza się po środowisku mBlock, wie, gdzie szukać bloków do tworzenia skryptów sterujących różnymi czujnikami robota,
- potrafi stworzyć prosty skrypt sterujący czujnikami robota,
- potrafi wykorzystać parametr czujnika jako element instrukcji warunkowej,
- umie wykorzystać pomiar z czujnika do sterowania innymi częściami robota,
- wie, jak wykorzystać czujniki, aby stworzyć interaktywnego robota,
- potrafi stworzyć programy z wieloma funkcjami.

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 min)

STOPIEŃ TRUDNOŚCI/SKOMPLIKOWANIA

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

Od 2 do 5, w zależności od liczby wykonanych zadań

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,
- przygotować początkowe scenariusze dotyczące programowania robota mBot, tak aby ułatwić sobie prowadzenie zajęć.

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 czujnikami, 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 – 10 minut

Cel: wprowadzenie uczniów i uczennic w tematykę zajęć, omówienie zadań do wykonania. Na tych zajęciach uczniowie i uczennice wykorzystają wiedzę zdobytą na wcześniejszych lekcjach lub zajęciach dotyczących programowania i zaprogramują interaktywnego robota reagującego na otoczenie. Dzięki temu poznają zastosowania czujników i utrwalą wiedzę dotyczącą ich programowania.

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.

Dzielimy tablicę na 4 części – każda z części odnosi się do jednego kierunku poruszania się robota. Np. jedna z części będzie oznaczona jako „jazda do przodu”. Pytamy uczniów, w jakie funkcje chcieliby wyposażać swojego robota – takie, które można uruchomić za pomocą dostępnych czujników. Funkcje przypisujemy poszczególnym kierunkom, np. jadąc **do przodu**, robot mógłby wykonywać daną funkcję, a jadąc do tyłu, inną. Pomysły zapisujemy na tablicy w odpowiednich rubrykach. Jeśli wiemy, że którąś z funkcji trudniej będzie zrealizować, radzimy uczniom, aby najpierw wykonali łatwiejsze zadania (czyli wybrali prostsze funkcje).

Dodatkowo, mamy też przygotowane własne pomysły, które mogą uprościć realizację zadań, i w razie potrzeby dodajemy je do pomysłów uczniów. Poniżej przykładowe pomysły.

Jadąc do przodu, robot może:

- ⇒ reagować dźwiękiem na przeszkody: gdy robot zbliży się do przeszkody, wyda sygnał dźwiękowy, a gdy dotrze do krawędzi, cofnie się i zawróci;
- ⇒ zapalać światła jazdy dziennej;
- ⇒ zapalać światła i emitować sygnały na wzór radiowozu policyjnego.

Jadąc do tyłu, robot może:

- ⇒ wydawać sygnał ostrzegawczy dźwiękowy lub świetlny.

Skręcając w lewo lub w prawo, robot może:

- ⇒ sygnalizować kierunek jazdy zapalając światła (kierunkowskaz).

Uczniowie mogą wykonać zadania w dowolny sposób. Aby zaliczyć program zajęć, należy w swoim robocie zaprogramować funkcję poruszania się oraz dodać po jednej

funkcji dla każdego kierunku jazdy: do przodu, do tyłu, skręt w prawo, skręt w lewo.

Omówienie sposobu realizacji zadań – 5 minut

Cel: pokazujemy uczniom „szkielet” programu, do którego będą dodawać poszczególne funkcje. Aby ułatwić uczniom pracę i pozwolić im skupić się na realizacji zadania, prezentujemy schemat bloków, do których można dodawać kolejne elementy. Bardziej zaawansowane grupy mogą same spróbować stworzyć taki schemat i zaprezentować go całej grupie.

Na poniższym rysunku pokazany jest prosty skrypt z dodanym nowym blokiem „klawisz... naciśnięty?”, z kategorii „Czujniki”. Można go wykorzystać jako element instrukcji warunkowej.



Program tworzymy dla każdego z czterech kierunków. Działa on następująco: gdy klikniemy w „zieloną flagę”, rozpocznie się realizacja programu w pętli „zawsze” (czyli program będzie działał bez przerwy po kliknięciu flagi).

Jeśli zostanie spełniony warunek naciśniętego przycisku, realizowany będzie program, który „wstawimy” pod spodem. Jeśli nie (tzn. klawisz nie zostanie naciśnięty), będzie realizowany inny program. W tym przypadku będzie to zatrzymanie silników robota.

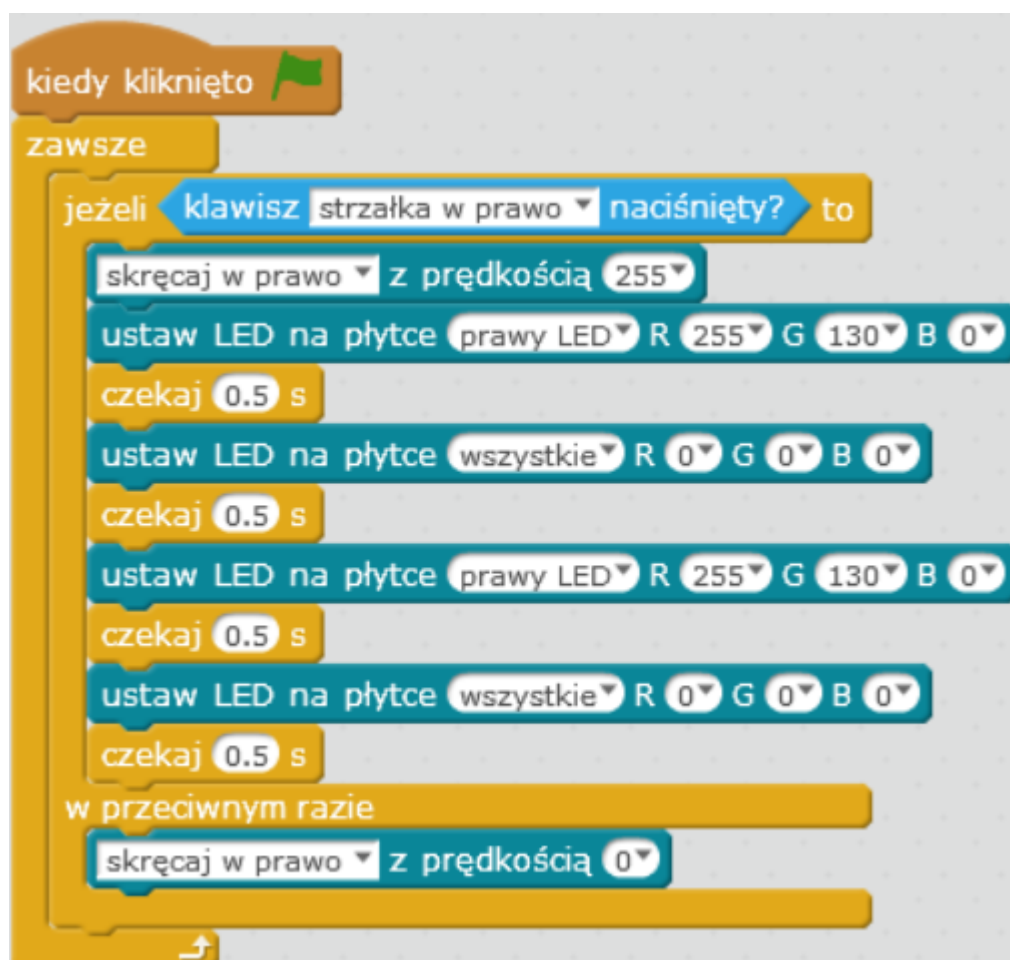
Proponowany sposób realizacji programu: jeśli klawisz „strzałka w prawo” zostanie naciśnięty, robot będzie poruszał się i zrealizuje daną funkcję, a jeśli nie, robot nie poruszy się. Na rozgrzewkę (w pierwszej części zajęć, do przerwy) najlepiej będzie rozpocząć od skręcania robota oraz cofania, ponieważ są to funkcje najprostsze.

Realizacja funkcji – 30 minut

Cel: uczniowie samodzielnie dodają wybraną funkcję do swojego robota. Realizują zadanie w dowolny sposób, pamiętając o warunkach zaliczenia zajęć.

Warto przypomnieć uczniom, aby sprawdzali działanie każdej stworzonej funkcji. Można też zapisywać poszczególne etapy (i proponowane skrypty), aby w razie potrzeby lub problemów móc do nich wrócić. Jazda robota do przodu oferuje najwięcej możliwości, więc zalecane jest pozostawienie jej na koniec.

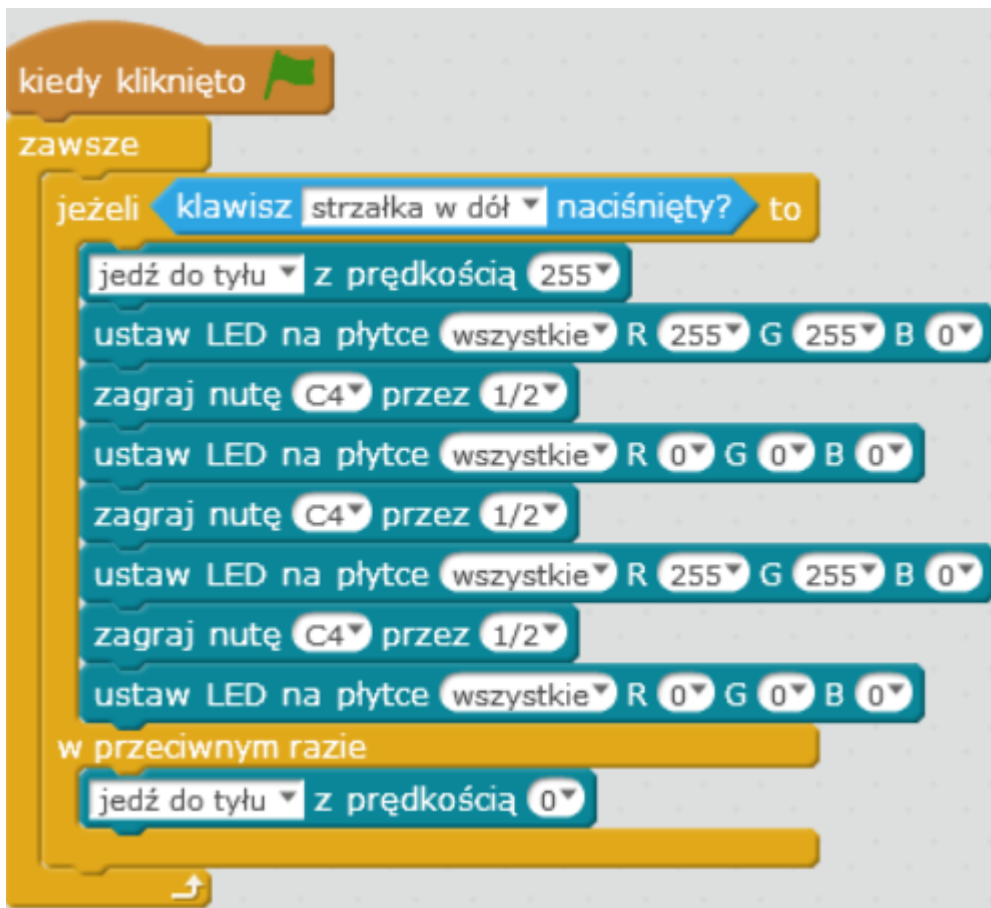
Poniżej zostały zaprezentowane przykładowe programy wraz z ich omówieniem. Jeśli uczniowie mają własne pomysły, możemy pokazać im te programy jako przykłady. Jeśli uczniowie zrealizują proponowane zadania, wykorzystujemy zaprezentowane skrypty do sprawdzenia ich działania lub do wspierania uczniów w ich pracy.



Powyższy program umożliwi robotowi skręt w prawo. Gdy zostanie naciśnięty klawisz „strzałka w prawo”, robot będzie skręcać w prawo, sygnalizując to diodą LED migającą w kolorze pomarańczowym. Dioda będzie naprzemiennie zapalać się i gasnąć. Gdy klawisz nie będzie wciśnięty, nastąpi wyłączenie ruchu silnika. Parametry kolorów możemy sprawdzić korzystając m.in. z następującej strony:

<http://www.kurshtml.edu.pl/generatory/kolory.html>.

Takie samo rozwiązanie można wprowadzić dla przeciwnego kierunku: skrętu w lewo.



Program zaprezentowany na powyższym rysunku „włącza” sygnalizację świetlną i dźwiękową robota podczas jego cofania. Zamiast opóźnień za pomocą bloków „Czekaj”, zostały tu wykorzystane bloki „zagraj nutę” o czasie trwania 1/2 sekundy.

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.

Dalsza część realizacji funkcji – 25 minut

Cel: uczniowie samodzielnie dodają do programu nowe funkcje.

W dalszej części zajęć można się skupić na poruszaniu się robota do przodu i dodać mu funkcje: włączanie świateł, ostrzeganie o przeszkodach, zawracanie w przypadku dojechania do krawędzi. Funkcje te możemy dodawać jako kolejne funkcje warunkowe „jeżeli”.



Robot, jadąc do przodu, sprawdza odległość od obiektów znajdujących się przed nim i sygnalizuje dźwiękiem napotkanie przeszkody. Można tu dodać takie funkcje jak: emitowanie coraz wyższego dźwięku, gdy przeszkoda znajduje się coraz bliżej. Jeśli robot dojedzie do krawędzi (lub czarnej linii), samodzielnie zawróci. Czasy („czekaj... s”) można dobrać dowolnie (np. 2 sekundy), a dodatkowo można po bloku „jedź naprzód” dodać oświetlenie drogi.

Wykonując nasz program możemy napotkać trudności z płynnością działania robota, ponieważ bloki „wykonują się” po kolei. Robot musi wykonać jedną czynność, aby mógł wykonać następną.

Na wcześniejszych zajęciach nie był wykorzystywany czujnik światła, ale uczniowie mogą dodać taką funkcję do swoich programów (np. jako czujnik zmierzchu lub do

zmiany jasności kierunkowskazów – zależnie od pory dnia).

Prezentacja prac uczestników – 10 minut

Cel: podsumowanie zajęć i prezentacja funkcji robota.

Każdy z zespołów prezentuje działanie swojego robota. Najciekawsze zrealizowane funkcje mogą omówić sami uczniowie. Możemy sprawdzić, ile funkcji udało się zrealizować spośród tych zapisanych na tablicy. Jeśli którejś z nich nie udało się wykonać, możemy zapytać, jakie problemy pojawiły się podczas realizacji. Uczniowie mogą opowiedzieć, w jaki sposób zaprogramowali daną funkcję i jak ona działa. Programy mogą zostać także zapisane w formie obrazka i powieszone w sali, tak aby można było wykorzystać je na przyszłych zajęciach. W przyszłości mogą też pojawić się nowe pomysły, które można będzie dodać do tak zapisanych programów.

Uczestnicy zapisują swoje programy i wyłączają roboty.

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 ZAJĘĆ:

Uczeń / uczennica, pracując samodzielnie albo w dwu- lub trzyosobowym zespole tworzy skrypt sterujący robotem mBot i jego czujnikami. Za pomocą stworzonego skryptu steruje robotem tak, aby robot poruszał się w różnych kierunkach i wykonywał inną (przynajmniej jedną) funkcję w czasie jazdy (lub skrętu) w danym kierunku. Zespoły prezentują swoje funkcje na forum grupy.

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 opóźnić wykonanie skryptu.

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

Dodatkowe materiały - linki:

Nauka programowania w języku i środowisku Scratch – zasoby edukacyjne programu „Mistrzowie Kodowania”:

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

Nauka programowania w języku i środowisku Scratch – zasoby edukacyjne programu „Koduj z Klasą”:

http://kodujzklasa.ceo.org.pl/sites/kodujzklasa.ceo.org.pl/files/scenariusze_scratch.zip

Nauka programowania w języku i środowisku Scratch – zasoby edukacyjne programu „Link do Przyszłości”:

www.linkdoprzyszlosci.pl/zasoby

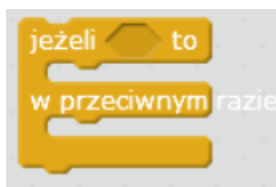
Nauka programowania w języku i środowisku Scratch – zasoby edukacyjne programu „#SuperKoderzy”:

<http://superkoderzy.pl/scenariusze-lekcji/podstawy-scratcha/>

Nauka programowania w środowisku mBlock z wykorzystaniem robota mBot - zasoby edukacyjne programu „Mistrzowie Kodowania”:

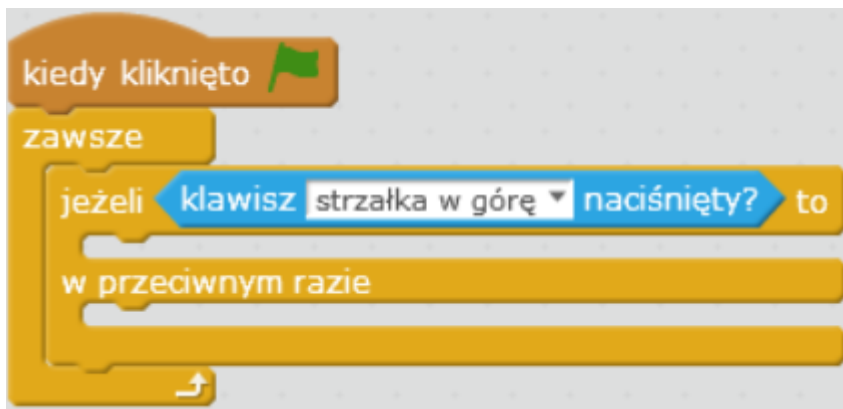
http://wiki.mistrzowiekodowania.pl/index.php?title=Pierwsze_kroki_-_zdalne_sterowanie_robotem

Blok dla instrukcji warunkowej jest przedstawiony na poniższym rysunku:

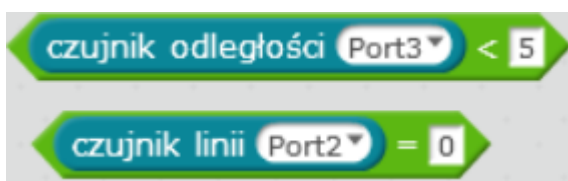


Jeśli spełniony jest dany warunek, wykonywany jest blok z pierwszego pola, jeśli nie – blok z drugiego pola.

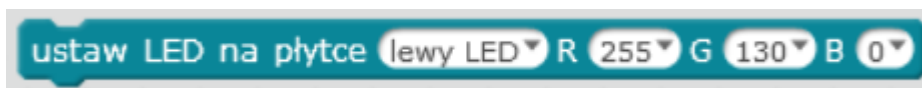
Szablon do realizacji funkcji wykorzystujące instrukcję warunkową:



Warunek sprawdzający odległość od przeszkody oraz warunek sprawdzający krawędź lub czarną linię.



Sterowanie diodami LED.



Tester kolorów (strona, na której można łatwo dobrać parametry koloru, który chcemy wykorzystać do świateł robota):

<http://www.kurshtml.edu.pl/generatory/kolory.html>

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/).