

# PROGRAMOWANIE W NAUCZANIU ZINTEGROWANYM

Jolanta Sobera, Anna Szczerba-Zubek, Natalia Cieślak

Uniwersytet Śląski

ul. Bankowa 14, 40-007 Katowice

[jolanta.sobera@us.edu.pl](mailto:jolanta.sobera@us.edu.pl); [www.math.us.edu.pl/pdm/sobera/](http://www.math.us.edu.pl/pdm/sobera/)

[anna.szczerba-zubek@us.edu.pl](mailto:anna.szczerba-zubek@us.edu.pl); [www.math.us.edu.pl/pdm/azubek/](http://www.math.us.edu.pl/pdm/azubek/)

[natalia.cieslar@us.edu.pl](mailto:natalia.cieslar@us.edu.pl); [www.math.us.edu.pl/pdm/cieslar/](http://www.math.us.edu.pl/pdm/cieslar/)

*Abstract. The aim of this article is to show how programming elements in early school could be introduced. Moreover we would like to reveal the combination of computer science with film and literature.*

## 1. Wstęp

Nauka programowania bardzo dobrze wpisuje się w strategiczne cele związane z rozwojem kapitału ludzkiego i społecznego Polski, wyrównywaniem szans osób z różnych środowisk i regionów oraz budowaniem gospodarki konkurencyjnej opartej na wiedzy. Tak naprawdę kształcenie w tym kierunku jest jednym z głównych warunków, które mogą zagwarantować rozwój innowacji technologicznych, biznesowych oraz społecznych. Umiejętność programowania i myślenie komputacyjne, czyli umiejętność kreatywnego rozwiązywania problemów z różnych dziedzin metodami wywodzącymi się z informatyki, są ważne nie tylko ze względu na rosnące zapotrzebowanie na rynku pracy w Polsce, Europie i na świecie. Myślenie komputacyjne wiąże się także z innymi kluczowymi kompetencjami XXI wieku, takimi jak określanie samego problemu, rozkładanie problemu na elementy składowe, znajdowanie prawidłowości i reguł, eliminowanie tego co nie istotne, tworzenie rozwiązań krok po kroku, znajdowanie i poznawanie błędów. Kompetencje te są w sumie niezbędne, aby aktywnie uczestniczyć w dynamicznie zmieniającym się środowisku życia zawodowego i społecznego.

W edukacji wczesnoszkolnej nauczanie kodowania/programowania powinno wprowadzać się poprzez zabawę wykorzystując różnorodne środki dydaktyczne, w tym gry planszowe, klocki Numicon (tworzenie kompozycji według wzoru), klocki Dienes'a (określanie cech przedmiotów takich jak: wielkość, kolor, kształt, grubość itp.), zabawki, które można zaprogramować. Wykorzystanie tych środków dydaktycznych ma na celu, aby programowanie nie odbywało się wyłącznie z użyciem

tabletów czy komputerów. Modelowanie w trakcie gry i zabawy zachowań, które polegają na rozwiązywaniu prostych problemów sprawiają, że od najmłodszych lat dzieci w naturalny sposób zaczynają funkcjonować w środowisku algorytmicznym. Równocześnie proponuje się rozszerzyć naukę programowania o zajęcia z wykorzystaniem robotów. Obecność robotów na zajęciach z informatyki powoduje wyraźny wzrost zainteresowania dzieci w różnym wieku nauką programowania.

Kolejnym ważnym elementem, na który należy zwrócić uwagę jest, aby na wszystkich etapach kształcenia, współpraca nauczycieli różnych przedmiotów była prawdziwa i realna. Chodzi o to, aby zajęcia z programowania łączone były z rozwiązywaniem prawdziwych problemów z innych dziedzin nauki i życia. Nauka programowania powinna obejmować „pełny” proces – od pomysłu i wymagań funkcjonalnych, przez testowanie, po instrukcję dla użytkownika, itp. Wszystko to stwarza możliwość i konieczność zorganizowania pracy w grupie, lepiej odzwierciedla „biznesową” rzeczywistość oraz ukazuje rangę samego pomysłu, który stanowi podstawę udanych start-upów. Duże znaczenie mają projekty interdyscyplinarne z technologiami informacyjnymi „w tle”, które angażują cały „ekosystem” szkoły, z uwzględnieniem rodziców, lokalnych instytucji, w tym bibliotek i mediów. (Całość Wstępu jest oparta na [3]).

Celem niniejszego artykułu jest pokazanie jak można wprowadzać elementy programowania w nauczaniu wczesnoszkolnym a także ukazanie połączenia zajęć z informatyki z innymi przedmiotami.

## 2. Dziecko badaczem przyrody

Zwykle informatyka jest kojarzona z fizyką, matematyką czy, na wczesnym etapie edukacyjnym, z zajęciami z przyrody. Pytanie czy można skorelować informatykę z zajęciami humanistycznymi. W jedną stronę jest łatwo, na zajęciach humanistycznych można podejmować temat o komputerach i ich możliwościach o zagrożeniach z nimi związanych. W literaturze znajdziemy wiele pozycji związanych z tą tematyką „Bajki robotów” Lema czy popularne wśród młodzieży „Feliks Net i Nika” Rafała Kosika. Ale czy na lekcji informatyki wykorzystamy literaturę i film aby pobudzić słuchaczy do działania? W omawianych zajęciach bazujemy głównie na filmach. Pierwszy z nich to „Toy story”, które miało swoją premierę w 1995 roku – film animowany produkcji amerykańskiej stworzony przez firmę Pixar we współpracy z firmą Disney. Jest to pierwszy film pełnometrażowy stworzony całkowicie przy użyciu techniki komputerowej. Drugim filmem, który był inspiracją do zajęć to „Noc w muzeum” (ang. *Night at the Museum*) – amerykańska komedia fantastyczna z 2006 roku. Oczywiście nie może obejść się bez wspomnienia „Gwiezdnych Wojen” – amerykańskiej franczyzy filmowej z gatunku space-opera opartej na trylogii filmów stworzonych przez George'a Lucasa, której akcja rozgrywa się „dawno temu

w odległej galaktyce”. Przykładowe zajęcia są realizowane w Pałacu Młodzieży w Katowicach w ramach cyklu „Dziecko badaczem przyrody”. Uczestnikami są dzieci ze szkół podstawowych województwa śląskiego (także klas integracyjnych). Słuchaczami są uczniowie klas II-IV, czasem na prośbę nauczyciela uczestniczą w nich uczniowie klas starszych. Dzieci pracują 45 minut w parach lub grupach trzysobowych. Do zajęć niezbędne są zestawy LEGO EV3 lub LEGO NXT, komputer lub tablet (w przypadku EV3) z zainstalowanym oprogramowaniem LEGO.

Zajęcia rozpoczynamy od sprecyzowania celu. „Toy Story” zna ponad 90% uczniów, „Noc w muzeum” około połowa słuchaczy. Rozmawiamy o głównych bohaterach, o ich zachowaniu. Wszyscy zgadzają się, że eksponaty lub zabawki ożywają gdy jest ciemno. To jest punkt wyjścia, uczniowie wiedzą, że w czasie zajęć ich robot będzie działał gdy w sali będzie ciemno i przestanie pracować gdy zapalimy światło.

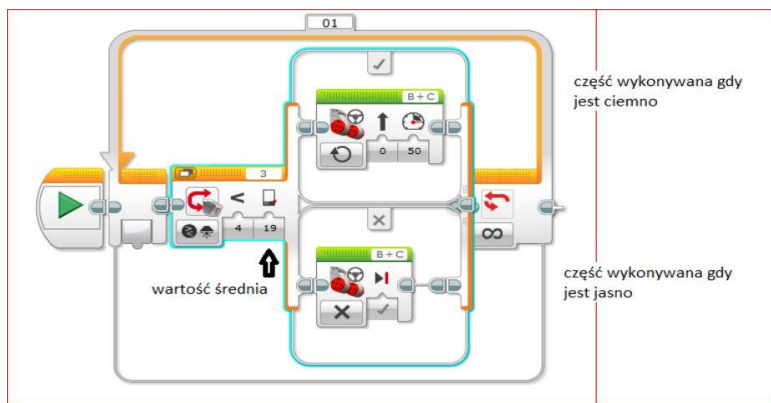
Dzieci otrzymują pojazd zbudowany z LEGO MINDSTORMS, dwa klocki, czujnik koloru i światła (Rysunek 1).



Rysunek 1

Pierwszym zadaniem jest uruchomienie „robota”. Uczestnicy sprawdzają (lub przypominają sobie – jeśli są na warsztatach po raz kolejny) podstawowe ustawienia ikony „Move steering”, tj. ustawienie prędkości, skręcanie, jazda do tyłu, hamowanie. Następnie słuchacze otrzymują polecenie przymocowania czujnika do robota w taki sposób, aby był skierowany do góry. Po zakończeniu budowy sprawdzamy wskazania czujnika gdy w sali jest jasno (zapalone światła) i ciemno (światło zgaszone, żaluzje zaciągnięte). Uzyskane wyniki zapisujemy na tablicy i w karcie pracy (rys. 9). Uczniowie poznają pojęcie średniej. W klasach starszych jest to średnia arytmetyczna z kilku wartości (od 4 do 6 gdyż zwykle tyle jest grup), w klasach młodszych uczniowie liczą średnią z dwóch wartości: największej z otrzymanych wyników gdy jest ciemno i najmniejszej z otrzymanych wyników gdy jest jasno. Wszyscy dzielą zbiór liczb (wyników testu) na dwa: mniejsze i większe od wyznaczonej wartości średniej. Uczniowie zauważają, że wartości mniejsze otrzymaliśmy, gdy było w sali ciemno, a większe – gdy było jasno. Może zdarzyć się tak, że jeden z wyników należy do zbioru innego niż powinien (ja miałam dwa takie przypadki na około 60 prowadzonych zajęć). Oczywiście należy zinterpretować zaistniałą sytuację i wskazać jej przyczyny. U mnie dzieci postawiły robota przy monitorze i gdy w sali było ciemno ich robot wykrywał jasne światło. Następnie sprawdzamy wskazania czujnika – gdy skierujemy go w stronę światła słonecznego okazuje się, że można uzyskać bardzo wysoką wartość (około 80 – czujnik oparty

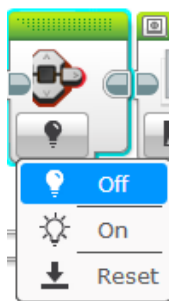
jest na skali od 0 (bardzo ciemno) do 100 (bardzo jasno)). Teraz przechodzimy do głównego celu zajęć jakim jest programowanie. Z zakładki pomarańczowej wybieramy ikony pętli (loop) oraz przełącznika (switch), który umieszczamy w środku pętli. Wpisujemy uzyskaną przez nas wartość średnią. Następnie programujemy pierwszą funkcjonalność jaką jest ruch (rys. 2).



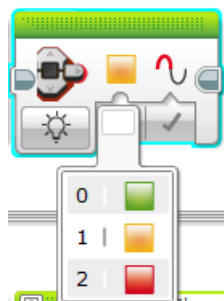
Rysunek 1. Program – etap 1

Uczniowie proponują inną funkcjonalność zabawki. Spośród zgłoszonych wybieramy te, które są możliwe do uzyskania ze względu na ograniczenia związane z budową i programowaniem.

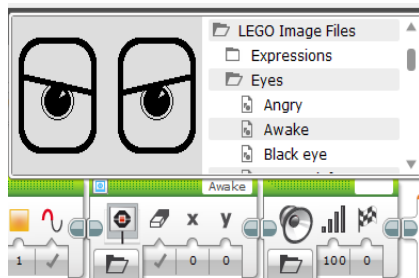
Zwykle wśród propozycji znajduje się: wydawanie dźwięków, skręcanie. Dodają do tego automatyczne zapalenie lampki kontrolnej (rys. 3 i 4) i wyświetlanie oczu stosowanych do sytuacji (zamkniętych lub otwartych – rys. 5). Przy wyborze dźwięków można zaproponować uczniom C3-PO lub A2D2 nawiązując do znanej serii „Gwiezdne wojny” (dźwięki Beep1-Beep4 – rys. 6). Cały program prezentuje rys. 7.



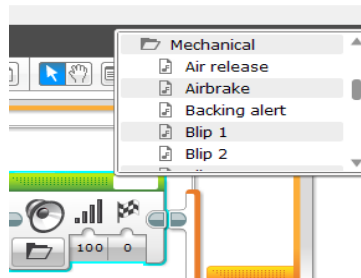
Rysunek 2 Wyłączenie lampki kontrolnej



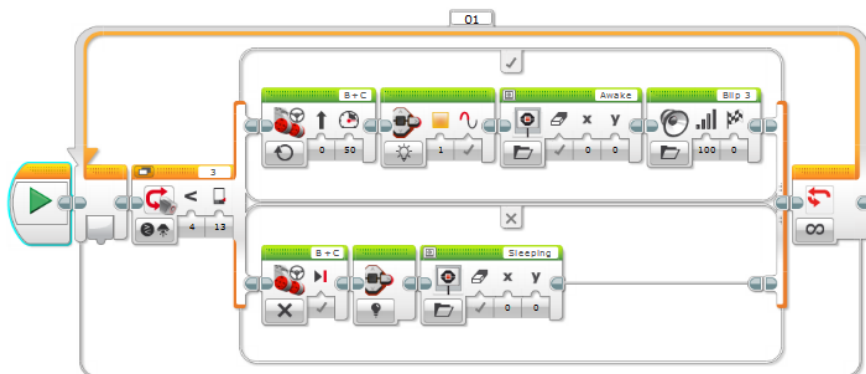
Rysunek 3: Ustawianie koloru



Rysunek 4 Wybór oczu dla robota



Rysunek 5 Wybór dźwięku dla robota



Rysunek 7 Cały program

Po zabawie pora podsumować zajęcia. Dzieci szukają zastosowań czujnika światła w swoim otoczeniu. Najczęstszą propozycją jest automatyczne zapalenie światła w pojazdach, czy lamp na ulicach.

Karta pracy – klasa IV

Imię i nazwisko .....

klasa .....

szkoła.....

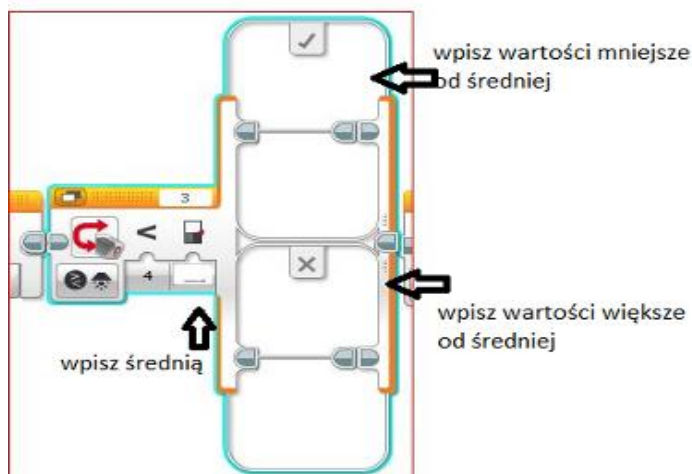
### Co zabawki robią w nocy?

W tabeli wpisz odczyty czujnika światła podane przez poszczególne grupy:

	Jasno	Ciemno
Grupa 1		
Grupa 2		
.....		

$$\bar{x} = \square + \square + \square + \square + \square + \square + \square = \frac{\square}{2} = \square$$

W tabeli kolorem żółtym pokoloruj te pola, w których wartość są większe od średniej a kolorem zielonym te, które są mniejsze od średniej.



Rysunek 8 Przykładowa Karta pracy dla grupy starszej

## Literatura

1. Materiały edukacyjne dla LEGO Mindstorms
2. Podstawa programowa z informatyki – szkoła podstawowa.
3. Rekomendacje w sprawie uczenia/się programowania i rozwijania kompetencji cyfrowych; [http://www.ceo.org.pl/sites/default/files/news-files/rekomendacje\\_w\\_sprawie\\_programowania\\_1.pdf](http://www.ceo.org.pl/sites/default/files/news-files/rekomendacje_w_sprawie_programowania_1.pdf)