

Majsterkowicze 2.0

Autor: Maciej Wojnicki, Grzegorz Zawistowski

Lekcja 5:

Jak sterować robotem z komputera?

Zapoznanie uczniów z innymi możliwościami Scratcha niż znane nam dotychczas programowanie obiektów na ekranie. Pokazanie, w jaki sposób można sterować robotem z poziomu komputera. Testowanie podstawowych bloków kontrolujących działanie sterownika LOFI Brain. Stosowanie podstawowych pojęć: funkcja, parametr, zmienna.

Cele zajęć:

Uczeń powinien:

- Rozumieć pojęcia: funkcja, parametr, zmienna, sterowanie, programowanie.
- Uruchamiać Scratcha on-line.
- Podłączyć za pomocą kabla USB sterownik LOFI Brain do komputera.
- Sterować silnikami pojazdu z poziomu Scratcha i zmieniać parametry silników (silnik, kierunek, moc).
- Tworzyć proste skrypty umożliwiające dynamiczne sterowanie pracą silników przy pomocy myszy i klawiatury.
- Tworzyć nowe zmienne i odczytywać wartości z czujników.

Pojęcia kluczowe:

→ Sterownik LOFI Brain → Blok / Funkcja → Parametr
→ Zmienna → Sterowanie → Programowanie

Czas na realizację zajęć: 45 minut (1 godzina lekcyjna)

Metody pracy:

- Wykład problemowy
- Dyskusja dydaktyczna związana z wykładem
- Pokaz
- Ćwiczenia przedmiotowe
- Projekt

Materiały pomocnicze:

- Laptop/komputer nauczycielski z zainstalowaną przeglądarką internetową Chrome i wtyczką LOFI Robot ScratchX Chrome* oraz wtyczką CODEBENDER
- Projektor i ekran projekcyjny.
- Komputery uczniowskie z zainstalowaną przeglądarką internetową Chrome i wtyczką LOFI Robot ScratchX Chrome oraz wtyczką CODEBENDER .
- Dostęp do internetu na wszystkich komputerach.
- Zestaw EDUBOX LOFI Robot - pojazd zmontowany w czasie lekcji 4, kable USB do podłączenia sterowników z komputerami lub moduły BLUETOOTH.
- Sterowniki LOFI Brain muszą mieć wgrany wcześniej Lofi FIRMATA. Instrukcja, jak wgrać Lofi FIRMATA na sterownik LOFI BRAIN (zarówno do komunikacji przez USB, jak i do komunikacji BLUETOOTH) oraz jak zainstalować wtyczkę do Chrome, dostępne są na stronie - <http://www.lofirobot.com/edubox/scratchx-chrome/>
- Jeśli edytor ScratchX się nie uruchamia należy sprawdzić czy na komputerach jest zainstalowana aktualna wersja FLASH PLAYERA - https://scratch.mit.edu/info/ext_download/

Treści programowe (związek z podstawą programową)

Podstawa programowa kształcenia ogólnego dla szkół podstawowych – II etap edukacyjny – klasy IV-VI. Zajęcia komputerowe. Treści szczegółowe:

1. Bezpieczne posługiwanie się komputerem i jego oprogramowaniem. Uczeń:
 - 1.5. posługuje się podstawowym słownictwem informatycznym;
5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera. Uczeń:
 - 5.1. za pomocą ciągu poleceń tworzy proste motywy lub steruje obiektem na ekranie;
 - 5.2. uczestniczy w pracy zespołowej, porozumiewa się z innymi osobami podczas realizacji wspólnego projektu, podejmuje decyzje w zakresie swoich zadań i uprawnień.



Przebieg zajęć:

1. Wprowadzenie w tematykę i integracja grupy

Czas na realizację tej części: ok. 10 minut.

Rozpoczynamy dyskusję, zadając uczniom pytanie: Czy poznaliście już program Scratch?

Słuchamy odpowiedzi. Uczniowie powinni już znać program, ponieważ w ramach programu #SuperKoderzy pierwsze 4 lekcje dotyczyły Scratcha.

Zadajemy uczniom pytanie: Co może robić w programie Scratch?

- Tworzyć własne animacje.
- Tworzyć własne gry komputerowe.
- Tworzyć programy edukacyjne.
- Projektować multimedialne, interaktywne kartki świąteczne itp.

Na komputerze nauczyciela uruchamiamy przeglądarkę internetową i aplikację ScratchX

(http://www.lofirobot.com/scratchx/?url=http://lofirobot.github.io/Scratch/lofirobot_chrome.js#scratch).

Wyświetlamy obraz z komputera nauczycielskiego na tablicy.

Zadajemy kolejne pytanie: Czy wiecie, jak podzielone jest okno programu Scratch?

Jest to okazja do dyskusji i pokazania przy pomocy projektora oraz omówienia budowy programu i zasady działania (nauczyciel pokazuje, daje wypowiedzieć się uczniom - opisywać poszczególne elementy i sam dopowiada):

- Po lewej stronie widzimy okno z ikoną duszka symbolizującego naszego robota.
- W środkowej części, podzielone na kategorie, znajdują się polecenia czyli bloki. Możemy je klikać - wówczas zostaną natychmiast wykonane przez duszka. Można je też wyciągać na prawą stronę ekranu i łączyć w ciągi poleceń - (skrypty), wówczas duszek wykona wszystkie sklejone bloki za jednym razem, lub będzie wykonywał różne akcje, w zależności od wcześniej ustalonych warunków.
- Większość bloków, głównie z kategorii: Ruch, Wygląd, Dźwięk, Pisak, Zdarzenia, Kontrola, Czujniki i Wyrażenia, poznaliście już na początkowych lekcjach ze Scratchem. Te bloki nadal służą do sterowania i programowania duszka w części głównej programu.
- Dziś zajmiemy się zakładką "Więcej bloków".

Nauczyciel prosi uczniów o włączenie komputerów i uruchomienie ScratchX w przeglądarkach internetowych Chrome.

2. Część zasadnicza

Czas na realizację tej części: ok. 30 minut.

Nauczyciel wykonuje opisane niżej ćwiczenia. Pokazuje uczniom, co robi i omawia. Po zakończeniu pokazywania każdego ćwiczenia nauczyciel prosi, aby uczniowie w grupach wykonali podobne próby na swoich komputerach, ze swoimi robotami.

Ćwiczenie 1 - sterujemy kołami pojazdu (10 min)

Podłączamy sterownik LOFI Brain do komputera za pomocą kabla USB. W przeglądarce na stronie ScratchX dioda przy opisie rozszerzenia Lofi Robot świeci się na zielono, co oznacza że nasz sterownik (robot) jest dobrze

podłączony z komputerem i sterowanie z poziomu ScratchX już jest możliwe.

Gdyby światelko świeciło się na żółto oznaczałoby to, że na sterownik najprawdopodobniej nie jest wgrany skrypt LOFI Firmata (więcej informacji w materiałach pomocniczych na początku scenariusza i w materiałach dla nauczyciela na stronie **LOFI ROBOT**: <http://www.lofirobot.com/edubox/scratchx-chrome/>).

Jeśli światelko będzie czerwone, oznacza to, że nasza przeglądarka nie wykrywa rozszerzenia do Scratcha (więcej informacji w materiałach pomocniczych na początku scenariusza i w materiałach dla nauczyciela na stronie LOFI ROBOT: <http://www.lofirobot.com/edubox/scratchx-chrome/>).

Zakładka “Więcej bloków” zawiera bloki służące do sterowania robotem. Ich nazwy są dość proste. Każdy blok to jakaś funkcja. Bloki posiadają parametry (np. wybór silnika, wybór kierunku obrotu, wybór mocy silnika).

Pierwszy blok dotyczy sterowania silnikami. Jeśli uruchomię poniższą funkcję:



Nauczyciel wyciąga ten blok na środek i klika.

to silnik podłączony do portu M1, będzie się obracał w kierunku “przód” z mocą “100%”.

W sumie możemy zmienić 3 parametry tej funkcji:

- port - gniazdo, do którego podłączony jest silnik: M1, M2.
- kierunek: przód / tył.
- Moc - wpisujemy od 0 do 100, bez znaku procent, ale domyślnie pamiętamy, że jest to wartość procentowa, czyli 100 - oznacza pełną moc / prędkość, a 0 - oznacza zatrzymanie silnika. Wartości ujemne nie są przyjmowane.

Omawiając powyższe kwestie demonstrujemy to uczniom zmieniając różne parametry tej funkcji na komputerze nauczyciela i pokazujemy, jak zachowuje się robot podłączony do komputera nauczyciela. Następnie prosimy o przetestowanie tego przez uczniów.

Zwracamy uwagę, że silnik już po jednokrotnym wywołaniu tej funkcji, obraca się z daną prędkością cały czas, do momentu kiedy ten sam silnik otrzyma kolejne polecenie, np. obracania z inną prędkością, czy obracania w innym kierunku.

Chcąc zatrzymać silnik, musimy mu wydać polecenie “obracania się z mocą 0”.

Ćwiczenie 2 - dynamicznie sterujemy prędkością silnika (ok. 10 min.)

Mówimy uczniom, że możemy też dynamicznie zmieniać parametry tej funkcji. Na przykład jeśli z sekcji “Czujniki” wezmę blok “X myszy” (który sczytuje położenie wskaźnika myszy na ekranie) i wstawię ten blok jako parametr do powyższej funkcji w mocy, to w zależności od tego gdzie (lewo-prawo) będę ustawiał wskaźnik myszy w polu gry w Scratch, to silnik będzie zmieniał prędkość.

Należy jednak pamiętać, aby powyższy blok wstawić w pętlę „Zawsze”, którą znajdziemy w sekcji “Kontrola”. Wówczas po kliknięciu na taką funkcję (umieszczoną w pętli „Zawsze”) komputer zawsze sczytuje położenie wskaźnika myszy na osi X i zmienia parametr mocy, a więc zmieniamy prędkość silnika w zależności od ruchów myszą na ekranie.

Jeżeli chcemy sterować silnikiem przy pomocy klawiatury, to tworzymy ciągłe polecenia (łączyliśmy kilka bloków). Na przykład:

- Kiedy klawisz Spacja naciśnięty - Obracaj Silnik M1 w kierunku tył z mocą 100
- Kiedy klawisz Strzałka w górę naciśnięty - Obracaj Silnik M1 w kierunku przód z mocą 100

Uwaga: przed rozpoczęciem ćwiczenia warto roboty położyć na jakiejś podstawie, tak aby koła nie dotykały ławki. W przeciwnym wypadku roboty mogą spaść na podłogę.

Uwaga: ponieważ funkcja sterująca silnikami przyjmuje parametr mocy od 0 do 100, to tak naprawdę gdy wskaźnik myszy jest poniżej 0 na osi X, lub powyżej 100, funkcja przyjmuje minimalne i maksymalne wartości (0 i 100).

Po wykonaniu powyższych 2 skryptów, spacja będzie uruchamiała kręcenie silnika w tył, a strzałka w górę będzie uruchamiała kręcenie silnika w przód.

Do sterowania robotem-pojazdem za pomocą klawiatury powrócimy na kolejnej lekcji.

Ćwiczenie 3 - odczytujemy dane z czujników (ok. 5 min)

Gdy konstruujemy robota, nie tylko chcemy nim sterować z poziomu komputera, ale chcemy go zaprogramować tak, aby wykorzystywał swoje czujniki i sam w odpowiedni sposób reagował. Chcemy, aby czujniki sterowały aktuatorami.

W naszym robocie-pojedździe mamy zamontowany jeden czujnik: odległości. Możemy odczytać jego wartość na komputerze przy pomocy bloku "Czujnik odległości". W tym celu wyciągamy ten blok na środek i gdy klikniemy na niego, w chmurce przez chwilę pojawi się wartość. Możemy zbliżyć rękę do czujnika i klikać na ten blok oraz oddalać rękę i klikać. Komputer będzie wyświetlał przez chwilę różne wartości.



Jeśli jednak chcemy, aby ta wartość (tu: odległość) odczytywana i wyświetlana była cały czas, musimy w sekcji "Dane" stworzyć nową "Zmienną". Możemy ją nazwać np. "odległość". Pokażą się bloki dotyczące tej zmiennej, a jej wartość wyświetla się w lewym górnym rogu programu. Aby korzystać z tej funkcji, należy napisać następujący skrypt:



W pętłę zawsze, wstawiamy polecenie "Ustaw ODLEGŁOŚĆ na" i jako parametr wstawiamy blok "Czujnik odległości".

Uwaga: elektroniczny czujnik odległości podaje swoje pomiary wolniej niż Scratch wykonuje pętlę ZAWSZE, dlatego musimy zwolnić częstotliwość jej powtarzania dodając opóźnienie. czyli blok "czekaj 0.1s"

Gdy uruchomimy (klikniemy zieloną flagę) ten skrypt, na ekranie cały czas pojawiać się będzie odczyt z czujnika odległości. Sprawdzamy to przysuwając i oddalając dłoń od czujnika naszego pojazdu.

Prosimy, aby uczniowie zrobili to samo i w między czasie mówimy uczniom, że w ten sam sposób działa blok: "Odczytaj wejście INPUT 1-4". W ten sposób, gdy do różnych wejść robota będziemy podłączać różne czujniki,

możemy odczytywać ich wartości i wstawiać jako parametry zmienne do funkcji.

odczytaj wejście INPUT 1 ▼

Zwracamy uwagę, że cztery pierwsze bloki z listy różnią się kształtem od bloków “odczytaj wejście” i “czujnik odległości”. Nie jest to przypadek, ponieważ różnią się one sposobem działania.



Pierwsze cztery bloki (obracaj silnik, ustaw wyjście na wartość, ustaw wyjście jako, ustaw buzzer) to bloki funkcyjne. Po ich uruchomieniu wykonane zostanie jakieś działanie np. uruchomienie silnika lub zapalenie diody. Kształt bloków funkcyjnych ma charakterystyczne wcięcia dzięki, którym można je łączyć ze sobą tworząc bardziej zaawansowane funkcje.

Bloki “Odczytaj wejście” i “czujnik odległości” mają okrągły kształt, są to tzw. bloki raportujące. Ich uruchomienie nie spowoduje żadnego widocznego działania. Blok raportujący sprawdza jakiś parametr, w tym wypadku mówi nam, jakie są wskazania czujnika podłączonego do któregoś z WEJŚĆ, a następnie pozwala nam przekazać tę wartość gdzieś dalej np. do któregoś z bloków funkcyjnych jako parametr. Przy ich pomocy możemy odczytać sygnały z czujników (z przycisku, potencjometru, czujnika światła) podłączonych do gniazd INPUT.

Ćwiczenie 4 - sterujemy wyjściami i głośnikiem-buzzerem (ok. 5 min.)

Teraz poświęćmy uwagę kolejnemu blokowi: “Ustaw wyjście OUTPUT 1-4 na wartość 0-100”

ustaw wyjście OUTPUT 1 ▼ na wartość 100 %

Przy pomocy tego bloku możemy sterować poziomem sygnału na WYJŚCIACH - OUTPUT 1-4, dzięki czemu możemy kontrolować np. jasność diody lub wskazania miernika napięcia. Sprawdzamy działanie tego bloku podłączając do wyjścia OUTPUT1 moduł diody LED.

Podobnie działa blok “Ustaw BUZZER jako WŁĄCZONY/WYŁĄCZONY”

ustaw BUZZER jako włączony ▼

Jest to blok funkcyjny sterujący wbudowanym w płytce LOFI Brain buzzerem, czyli małym piszczącym głośniczkiem. Posiada on jeden parametr określający stan działania buzzera jako WŁĄCZONY lub WYŁĄCZONY.

W zależności od ilości pozostałego czasu pozwalamy uczniom przetestować działania poznanych bloków.

Prosimy o odłączenie robotów od komputerów. Roboty składamy do pudełek. Wyłączamy komputery.

3. Podsumowanie i ewaluacja

Czas na realizację tej części: ok 5 minut.

W ramach podsumowania przypominamy uczniom, że każdy blok w Scratchu to jakaś funkcja, czyli polecenie wykonujące jakąś czynność (np. włączające silnik). W przypadku wielu funkcji możemy ustawiać parametry (jak nr silnika, moc, kierunek obrotu, rodzaj wyjścia itp.). W przypadku niektórych parametrów możemy wykorzystać zmienne, np. wówczas gdy odczytujemy wartości z czujników (z czujnika odległości, czujnika natężenia światła czy potencjometru).

Przypominamy, że ze sterowaniem mamy do czynienia najczęściej w przypadku maszyn, a roboty najczęściej programujemy. Oczywiście robotem możemy też sterować. Sterowanie polega zazwyczaj na wydawaniu pojedynczych poleceń, a maszyna (np. silnik) natychmiast je wykonuje. Programowanie polega na układaniu ciągów poleceń, np. w Scratchu na łączeniu kilku bloków (funkcji) w ciągi poleceń, które możemy nazwać skryptami, czyli programami.

Uwagi/alternatywy:

W scenariuszach lekcji w ramach ścieżki Majsterkowicze 2.0 w programie #Superkoderzy wykorzystujemy zestaw **EDUBOX** firmy LOFI Robot. Szkoły, które nie posiadają zestawu tej firmy scenariusze lekcji mogą zrealizować pracując na Arduino i zakupionych indywidualnie podzespołach elektronicznych (czujnikach, silnikach, diodach, płytkach prototypowych) lub innych zestawach robotów edukacyjnych.

Realizując lekcję, pamiętajmy, by zbierać dokumentację projektu. Można robić zdjęcia, filmy, spisywać najciekawsze odpowiedzi uczniów.