| **TEMAT:** Programowanie robotów - poruszanie się |
| --- |

| **SCENARIUSZ LEKCJI** | | | |
| --- | --- | --- | --- |
| ***Szkoła:*** | | ***Czas (minuty)***: | 90 |
| ***Nauczyciel:*** |  | ***Wiek uczniów:*** | 13-14 |

| ***Zagadnienie główne:*** | Jak zaprogramować robota do poruszania się |
| --- | --- |

| ***Tematyka:*** |
| --- |
| * Programowanie robota do poruszania się |
| ***Cele:*** |
| * Nauka programowania robota do poruszania się |
| ***Oczekiwane efekty:*** |
| * Wiedza, jak napisać program, aby robot się poruszał |
| ***Formy pracy:***   * praca w parach, praca w grupach   ***Metody:*** |
| * prezentacja, dyskusja, ćwiczenia interaktywne |

| **TOK LEKCJI** |
| --- |
| **Przebieg zajęć** |
| **WPROWADZENIE**  **Przedstawienie celu głównego lekcji:**  Programowanie ARTIEbota po raz pierwszy i odkrywanie, jak go wprawić w ruch  Przyznajmy, roboty są świetne. W tej lekcji przedstawiamy krok po kroku, łatwy do wykonania samouczek (z przykładami kodu), który prowadzi Was przez proces programowania podstawowego autonomicznego robota mobilnego, aby mógł się poruszać. |
|  |
| **CZĘŚĆ GŁÓWNA**  Nauczyciel pomaga uczniowi w napisaniu kodu złożonego przez nas robota i umożliwieniu mu poruszania się.  Krok 1 (obie opcje): Przejdź do strony: http://mindplus.cc/download-en.html i pobierz wersję programu odpowiednią dla systemu operacyjnego Twojego komputera. Zainstaluj program i uruchom Mind+. Po uruchomieniu, przełącz się w tryb Offline.    Jeśli pracujesz z Arduino, przejdź bezpośrednio do kroku 2. ->  Maqueen Plus:  Kliknij na "Extensions" (Rozszerzenia) i w zakładce "**Board**" (Płyta) wybierz "**micro:bit**", w zakładce "Shield" wybierz "**Maqueen Plus**" lub "**Maqueen Plus V2**", a w zakładce "**Sensor**" wybierz "**HUSKYLENS AI Camera**". Kliknij na przycisk "Back" (Wstecz), aby wrócić, i oprogramowanie jest gotowe do użycia wybranych modułów.  Podłącz mikro:bit do komputera za pomocą kabla USB micro, a wskaźnik zasilania się zaświeci.  Kliknij na przycisk "Connect Device" (Podłącz urządzenie) i wybierz "micro:bit". Jeśli konieczne, zainstaluj sterowniki urządzenia.  Przetestuj sekwencję przesyłania za pomocą tego kodu:  Slika na kojoj se prikazuje stol  Opis je automatski generiran  Kliknij przycisk "Upload" (Prześlij).  Robot powinien poruszać się do przodu przez jedną sekundę, a następnie zatrzymać się.  Wypróbuj tę sekwencję - robot powinien poruszać się zgodnie z opisem w komentarzach.    Kliknij przycisk "Upload" (Prześlij).  Robot powinien poruszać się zgodnie z opisem w komentarzach (żółte bloki).  Spróbuj różnych prędkości, aby sprawić, byTwój robot poruszał się szybciej lub wolniej.  Krok 2: Otwórz zakładkę "Extensions" (Rozszerzenia) i wybierz płytkę - Arduino UNO.  Krok 3: Przejdź do zakładki "Sensor" (Czujnik) i wybierz czujnik - HUSKYLENS AI Camera.  Krok 4: Po wyborze czujnika, kliknij na przycisk "<- Back" (Wstecz), a następnie jesteś gotowy do użycia bloków Arduino i Sensor. Przejdźmy do testu, aby sprawdzić, czy wszystko działa. Przed tym musisz podłączyć urządzenie. Podłącz swoją płytkę Arduino UNO za pomocą kabla USB i wybierz odpowiedni port COMX-Uno (lub CH340), w zależności od producenta Twojej płytki Arduino.  **Podstawowy ruch**  Pamiętasz tę tabelę ze scenariusza wprowadzenia sprzętu?   | Funkcja | Silnik A |  | | --- | --- | --- | | Kierunek | Digital 12 | Digital 13 | | Prędkość (PWM) | Digital 3 | Digital 11 | | Hamulec | Digital 9 | Digital 8 | | Odczyt prądu | Analog 0 | Analog 1 |   Dwa silniki (A i B) są lewym i prawym silnikiem. Wyprowadzenia cyfrowe 12 i 13 są używane do zmiany kierunku (HIGH - jeden kierunek, LOW - przeciwny kierunek), a wyprowadzenia PWM 3 i 11 są używane do ustawiania prędkości (0-255). Wyprowadzenia 9 i 8 włączają/wyłączają hamulce (HIGH - hamulce włączone, LOW - hamulce wyłączone).  Poniżej możecie zobaczyć przykładowy kod z komentarzami po prawej stronie, które pomogą Wam zrozumieć, jak to działa.    Krok 1a: Naciśnij przycisk "Upload" (Prześlij), aby przesłać ten kod do płytki Arduino UNO.    Krok 2a: Bądź ostrożny, robot zacznie się poruszać natychmiast po zakończeniu przesyłania kodu! Robot powinien poruszać się do przodu przez jedną sekundę, a następnie zatrzymać się.  Rozwiązywanie problemów - jeśli robot:  - porusza się do tyłu - zamień miejscami czerwone i czarne przewody w terminalach obu silników.  - obraca się zgodnie z ruchem wskazówek zegara - zamień miejscami czerwone i czarne przewody w terminalach silnika B.  - obraca się przeciwnie do ruchu wskazówek zegara - zamień miejscami czerwone i czarne przewody w terminalach silnika A.  Graphical user interface, application  Description automatically generated  Teraz twój robot powinien poruszać się do przodu, przy użyciu stanu HIGH na pinach 12 i 13.  Istnieje także prostszy sposób programowania ruchu - będziemy używać naszych własnoręcznie wykonanych bloków zamiast powtarzać całą serię bloków ustawiających piny.  Krok 1b: Kliknij na grupę "My Blocks" (Czerwona).  Krok 2b: Kliknij przycisk "Make a Block" (Utwórz blok) i zmień "nazwę bloku" na "Stop".  Krok 3b: Kliknij OK. |
| Masz już swój pierwszy blok, ale obecnie nie jest on jeszcze powiązany z żadnym działaniem, więc musimy go najpierw zdefiniować.  Slika na kojoj se prikazuje stol  Opis je automatski generiran  Blok "Stop" zawiera 4 bloki. Pierwsze dwa bloki ustawiają prędkość obu silników na 0, a ostatnie dwa bloki włączają hamulce. Jeśli chcesz użyć trybu bezhamulcowego, ustaw stan pinów cyfrowych 8 i 9 na LOW.  Teraz nadszedł czas, aby utworzyć blok wejściowy z 3 parametrami liczbowymi, które będą przechowywać prędkość obu silników oraz czas trwania. Kliknij ponownie przycisk "Make a Block" (Utwórz blok), a następnie kliknij "Add an input" (Dodaj wejście) - "number" - 3 razy, a rezultat powinien byc następujący:    Zmień nazwę bloku na "Drive", pierwszy parametr "float" na "SpeedA", drugi parametr "float" na "SpeedB", a trzeci parametr "float" na "Time". Następnie kliknij OK.    Podstawowym pomysłem jest pobranie wartości SpeedA i SpeedB (akceptowalny zakres: -255 do 255), sprawdzenie, czy którakolwiek z nich jest ujemna (lub obie), i jeśli tak - odwrócenie kierunku poprzez ustawienie odpowiedniego pinu kierunku. Jednak do sterowania silnikiem musisz używać wartości bezwzględnej na pinie PWM.    Spróbujmy teraz poruszyć robotem przy użyciu naszych bloków.  Ostatni blok "Stop" znajduje się w pętli nieskończonej, aby zakończyć ruch.  Zmień wartości w bloku "Drive", prześlij program na Arduino UNO i sprawdź, jak szybko porusza się Twój robot.  Teraz jesteś gotowy/-a do korzystania z ARTIEbota w bardziej zaawansowanych projektach, włączając w to korzystanie z kamery HuskyLens. |
| **PODSUMOWANIE**  Nie ma wątpliwości, że roboty są niesamowite. Kiedyś nawet mogą rządzić światem, ale miejmy nadzieję, że wtedy zlitują się nad swoimi biednymi miękkimi twórcami (czyli programistami robotów) i pomogą nam zbudować raj w kosmosie. To oczywiście żart, ale tylko trochę. |

| ***Metody*** | ***Formy pracy*** |
| --- | --- |
| ***prezentacja***  ***ćwiczenie interaktywne/symulacja komputerowa*** | ***praca w parach***  ***praca grupowa*** |

| ***Materiały:*** |
| --- |
| * <http://mindplus.cc/download-en.html> |

| ***Literatura*** |
| --- |

| **OBSERWACJE, UWAGI, NOTATKI** |
| --- |
|  |