

Medbot



Dowiedz się więcej o zachowaniach robotów, podczas transportu leków do szpitala z Medbotem!



Odkryj nowe praktyczne kompilacje i możliwości programowania, aby zrozumieć temat.

Wygląd gotowego Speedbota



GotowyVEX V5 Speedbot

Ten robot został zaprojektowany tak, aby można go było szybko zbudować i samodzielnie zaprogramować lub sterować nim za pomocą kontrolera V5.

Potrzebne elementy

Może zostać zbudowany z:

VEX V5 Classroom Starter Kit



Instrukcja budowy









Zielona ikona wskazuje, że konstrukcję należy odwrócić (do góry nogami).





Obecnie używany jest tylko jeden z dwóch podzespołów utworzonych w tym kroku. Drugi zostanie użyty później w kroku 9.



Upewnij się, że inteligentne silniki są ustawione we właściwym kierunku (otwory na śruby skierowane na zewnątrz konstrukcji, a otwór wału do wewnątrz).







Upewnij się, że silniki są ustawione we właściwym kierunku (otwory na śruby skierowane na zewnątrz konstrukcji, a otwór wału do wewnątrz).

















Zielona ikona wskazuje, że konstrukcję należy obrócić (o 180 stopni).





Niebieski znak pokazuje, jaka powinna być orientacja Mózgu robota, jeśli konstrukcja została odwrócona prawą stroną do góry. Upewnij się, że 3przewodowe porty w mózgu robota są skierowane w stronę Radia V5!





Zielone elementy wskazują, do którego portu w Mózgu robota należy podłączyć każde urządzenie za pomocą odpowiedniego kabla.

Wskazówki

Sprawdź Dodatek, aby uzyskać informacje na temat korzystania Hex Nut Retainers.

Analiza

Teraz, gdy skończyłeś budowę, włącz robota i zobacz, co robi! Obejrzyj swoją konstrukcję, a następnie odpowiedz na pytania w notatniku technicznym.

- Do jakich zadań ten robot byłby przydatny w prawdziwym świecie?
- Gdyby ten robot miał za zadanie realizować dostawy, jakie cechy byłyby ważne? Czemu?
- Jakie części robota chciałbyś zmienić lub ulepszyć, aby lepiej wykonać zadania z pytania 1? Wyjaśnij ze szczegółami lub szkicami.



Przetestuj swojego robota, obserwuj, jak działa, i wzmocnij swoją logikę i umiejętności rozumowania poprzez pomysłową, kreatywną zabawę.

Programowanie oparte na zachowaniach



Programowanie oparte na zachowaniach.

Złożoność programowania

Roboty można zaprojektować do wykonywania wielu różnych rodzajów zadań. Niektóre z tych zadań są bardzo proste, np. Otwieranie automatycznych drzwi. Inne nie są już takie proste, jak, np.: zaprogramowanie robota poruszającego się po magazynie, aby zebrać elementy do zamówienia. Bez względu na to, jak skomplikowane jest to zadanie, można je podzielić na prostsze elementy. Zadania te nazywane są zachowaniami i stanowią elementy składowe programowania robotów.

Zachowanie to sposób działania robota, które może wahać się od podstawowego do złożonego, w zależności od tego, jak robot jest zbudowany lub zaprogramowany. Prosty robot mobilny, taki jak Autopilot, ma tylko dwa silniki, więc jego zachowanie będzie obejmować obracanie tych silników, aby osiągnąć swoje cele. Dzięki większej liczbie projektów i kodu możesz zacząć od tego podstawowego zachowania i wykonywać coraz bardziej złożone zachowania.

Poniżej znajduje się lista zachowań robotów, od prostych do złożonych, zarówno dla Speedbota, jak i Clawbota. W nawiasach można zobaczyć prostsze zachowania, które składają się na każde z nich.

• Obróć silnik przypisany do określonego portu

- Jedź do przodu (obróć lewy i prawy silnik za pomocą układu napędowego)
- Przejedź 5 metrów (jedź do przodu i zatrzymaj się)
- Chwyć odległy obiekt (pokonaj 2 metry, obróć silnik kleszczy, aby go złapać)
- Podnieś przedmiot i połóż go na wysokiej półce (chwyć odległy przedmiot, obróć się, pokonaj 2 metry, użyj silników ramion i kleszczy, aby podnieść i zwolnić przedmiot)

Możesz zobaczyć, jak możesz zdekonstruować dowolne z bardziej złożonych zachowań na prostsze zachowania. Stają się one elementami składowymi każdego złożonego zadania.

Programowanie jazdy do przodu i do tyłu – bloki VEXcode V5

Speedbot jest gotowy do jazdy!

Te zajęcia dostarczą Ci narzędzi, dzięki którym będziesz mógł rozpocząć tworzenie ciekawych projektów dla Twojego Speedbota.

• Bloki VEXcode V5, które zostaną użyte:



• Aby dowiedzieć się więcej o bloku, otwórz Pomoc, a następnie wybierz blok jedź o.

№ 5	⊕ ▼ File 🤶 Tutorials	T VEXcode Project No	t Saved Controller BRAIN DOWNLOAD RUN STOP SHARE
Co	de		Help 🔲 🕗 🔪
Drivetrain	Drivetrain	when started	Drive
Looks	drive forward -	drive forward -	Moves the Drivetrain forever.
Events	drive forward - for 1 inches - >		drive forward 👻
Control	turn right -		
Sensing	turn right ← for 90 degrees ►		How To Use
Operators	turn to heading 90 degrees >		The drive block will run the Drivetrain forever, until a new Drivetrain block is used, or the program is stopped.
Variables	stop driving		Choose the direction the Drivetrain will move.
Comments	set drive velocity to 50 % -		
	set turn velocity to 50 % -	\odot	reverse
	set drive stopping to brake •		
	set drive timeout to 1 seconds		

 Upewnij się, że masz wymagany sprzęt, notes inżyniera i bloki VEXcode V5 pobrane i gotowe. Hardware/Software Required:

Amount	Hardware/Other Items	
1	Speedbot Robot	
1	Charged Robot Battery	
1	VEXcode V5 Blocks	
1	USB Cable (if using a computer)	
1	Engineering Notebook	

1. Preparing for the Exploration

Before you begin the activity, do you have each of these items ready? Check each of the following:

- Are the motors plugged into the correct ports?
- Are the smart cables fully inserted into all of the motors?
- Is the Brain turned on?
- Is the battery charged?

2. Start a New Project

Before you begin your project, select the Speedbot (Drivetrain 2-Motors, No Gyro) template project. The template project contains the Speedbot's motor configuration. If the template is not used, your robot will not run the project correctly.



Wykonaj następujące kroki:

- Otwórz menu Plik.
- Wybierz Otwórz przykłady.



- Wybierz i otwórz projekt szablonu Speedbot (Drivetrain 2-Motors, No Gyro).
- Ponieważ będziemy używać bloku *jedź / drive*, zmień nazwę swojego projektu na Drive.
- Zapisz swój projekt.
- Sprawdź, czy nazwa projektu Drive znajduje się teraz w oknie, pośrodku paska narzędzi.



3. Jazda do przodu



• Dodaj blok jedź do bloku po uruchomieniu w obszarze programowania.



• Na liście rozwijanej zmień jednostki z cali na milimetry.



• Zmień odległość z 1 mm na 100 mm.



• Kliknij ikonę **Slot.** Możesz zapisać swój projekt w jednym z czterech dostępnych miejsc w Mózgu robota. Kliknij cyfrę **1**.



• Podłącz robota do komputera lub tabletu. Ikona Mózg na pasku narzędzi zmienia kolor na zielony po pomyślnym nawiązaniu połączenia.



• Kliknij przycisk Pobierz na pasku narzędzi, aby pobrać projekt Drive do Mózgu.



- Sprawdź, czy projekt został pobrany do mózgu Speedbota, patrząc na ekran robota. Nazwa projektu powinna być wymieniona w slocie 1.
- Uruchom projekt na robocie, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk Uruchom w mózgu robota. Gratulujemy stworzenia pierwszego projektu!

4. Jedź do tyłu



- Zmień w bloku jedź o wyświetlanie na wstecz zamiast do przodu.
- Pobierz projekt.
- Uruchom projekt upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk *Uruchom* w Mózgu robota.

5. Poczekaj, a następnie jedź wstecz



- Dodaj blok *czekaj* przed blokiem *jedź o.* Dzięki temu robot będzie czekał przed rozpoczęciem jazdy do tyłu.
- Wstaw trzy sekundy do bloku *czekaj.* To nakazuje robotowi odczekać trzy sekundy, przed rozpoczęciem jazdy do tyłu.
- Pobierz projekt.
- Uruchom projekt, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk *Uruchom.*
6. Wyzwanie: ćwiczenie gry w koszykówkę!



Układ gry Basketball Drills

W wyzwaniu Basketball Drills robot musi być w stanie pokonywać szereg linii z różnych odległości. Robot przejedzie do przodu do pierwszej linii, która znajduje się 10 cm od pozycji wyjściowej, odczeka 1 sekundę, a następnie cofnie się, powracając do tej samej linii. Robot następnie powtórzy czynność, podjeżdżając 20 cm do przodu do drugiej linii, odczeka 1 sekundę, a następnie pojedzie wstecz do pierwotnej linii startu. Robot podjedzie do trzeciej linii w odległości 40 cm, odczeka 1 sekundę, a następnie wróci na linię startu, aby zakończyć wyzwanie.

Przed zaprogramowaniem robota zaplanuj ścieżkę i zachowanie robota w notatniku inżyniera.

Po ukończeniu wyzwania będziesz mógł łączyć ruchy do przodu i do tyłu z dodatkowymi zachowaniami robota, aby ukończyć jeszcze bardziej zaawansowane wyzwania.

Podczas programowania należy pamiętać, że 1 cm = 10 mm.

Programowanie skrętu w prawo i lewo – bloki VEXcode V5

Speedbot jest gotowy do skręcania!

Te zajęcia dostarczą Ci narzędzi, dzięki którym będziesz mógł rozpocząć tworzenie ciekawych projektów dla Twojego Autopilota.

• Bloki VEXcode V5, które zostaną użyte:



• Aby dowiedzieć się więcej o bloku, otwórz Pomoc, a następnie wybierz blok *skęć o*.

	Help	0	>
	Turn for		4
turn right - for 90 degrees >	Rotates the Drivetrain for a given distance.		
	turn right - for 90 degrees ►		
α an angle α an angle α at angle α			
	How To Use		
	Choose which direction the Drivetrain will turn.		
	turn right 🕶 for 90 degrees 🕨		
 A ADDALE A ADDALE A ADDALE 			

 Upewnij się, że masz wymagany sprzęt, notes inżyniera i bloki VEXcode V5 pobrane i gotowe do użycia.

llość	Sprzęt/inne przedmioty
1	Speedbot Robot
1	Naładowana bateria robota
1	Bloki VEXcode V5
1	Kabel USB (jeśli używasz komputera)
1	Notes inżyniera

1. Przygotowanie

Czy przed rozpoczęciem ćwiczenia masz gotowy każdy z tych elementów? Konstruktor powinien sprawdzić każdą z następujących kwestii:

- Czy kable są poprawnie włożone do wszystkich silników i czujników?
- Czy mózg / jednostka sterująca jest włączony/a?
- Czy bateria jest naładowana?

2. Rozpocznij nowy projekt

Przed rozpoczęciem projektu wybierz szablon Speedbot (Drivetrain 2-Motors, No Gyro) . Szablon Speedbot (Napęd) zawiera konfigurację silników i czujników Speedbot. Jeśli szablon nie jest używany, twój robot nie uruchomi projektu poprawnie.



Wykonaj następujące kroki:

• Otwórz menu Plik.

• Wybierz Otwórz przykłady.



- Wybierz i otwórz projekt szablonu Speedbot (Drivetrain 2-Motors, No Gyro).
- Ponieważ będziemy używać bloku skręć o / turn for, zmień nazwę swojego projektu na Turn.
- Zapisz swój projekt.
- Sprawdź, czy nazwa projektu **Turn** znajduje się teraz w oknie, pośrodku paska narzędzi.



3. Skręć w prawo



• Dodaj blok *skręć o* do bloku *po uruchomieniu* w obszarze programowania.



• Kliknij ikonę **Slot.** Możesz zapisać swój projekt w jednym z czterech dostępnych miejsc w Mózgu robota. Kliknij cyfrę **1.**



• Podłącz robota do komputera lub tabletu. Ikona Mózg na pasku narzędzi zmienia kolor na zielony po pomyślnym nawiązaniu połączenia.

• Kliknij przycisk Pobierz na pasku narzędzi, aby pobrać projekt Turn do Mózgu.

• Sprawdź, czy projekt został pobrany do mózgu Speedbota, patrząc na ekran robota. Nazwa projektu powinna być wymieniona w slocie 1 . Uruchom projekt na robocie, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk Uruchom w mózgu robota. Gratulujemy stworzenia pierwszego projektu!

4. Skręć w lewo

- Zmień parametr w bloku *skręć o* aby wyświetlać **lewą** stronę zamiast **prawej.**
- Pobierz projekt.
- Uruchom projekt, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk Uruchom.

5. Poczekaj, a następnie skręć w lewo

- Dodaj blok czekaj przed skręć o. Dzięki temu robot zaczeka, zanim skręci w lewo.
- Wstaw trzy sekundy w bloku *czekaj*. To nakazuje robotowi odczekać trzy sekundy przed skręceniem.

- Pobierz projekt.
- Uruchom projekt, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk Uruchom.

6. Ukończ wyzwanie Kamery Bezpieczeństwa!

Wyzwanie kamery bezpieczeństwa

W wyzwaniu kamery bezpieczeństwa robot musi zacząć od skrętu w prawo o 90 stopni. Następnie obróci się w lewo o 180 stopni, a później w prawo o 180 stopni "skanując", tak jak kamera bezpieczeństwa. Następnie robot powtórzy obrót o 180 stopni w lewo i 180 stopni w prawo jeszcze dwa razy. W ten sposób robot przeskanuje łącznie zakres 180 stopni trzy razy. Przed zaprogramowaniem robota zaplanuj ścieżkę i zachowanie robota w notatniku inżyniera.

Po ukończeniu wyzwania będziesz mógł łączyć ruchy do przodu i do tyłu z dodatkowymi zachowaniami robota, aby ukończyć jeszcze bardziej zaawansowane wyzwania.

Wprowadzenie do programowania tekstowego

Czym jest programowanie tekstowe?

Aby sterować robotem, będziesz tworzyć projekty w tekstowym języku programowania o nazwie VEXcode V5 Text. Jest to język oparty na C ++, używa tekstu i specjalnej składni do napisania instrukcji, które ostatecznie mówią robotowi, co ma robić. Jeśli wcześniej pracowałeś z programowaniem blokowym, takim jak bloki VEXcode V5, oznacza to, że instrukcje tekstowe zastępują bloki.

Instrukcja to cała linia w projekcie. Instrukcja może zawierać informacje o urządzeniu, komendzie, parametrach. Poniższy obraz przedstawia instrukcję z opisanymi każdej z tych części.

Należy pamiętać o konkretnych zasadach pisania instrukcji w tekście VEXcode V5. Na przykład wielkie litery mają określone zasady w instrukcjach. Urządzenia są pisane wielkimi literami, ale polecenia muszą w swojej składni rozpoczynać każdy wyraz wielka literą (*camel case*).

Aby uzyskać więcej informacji na temat formatowania gramatyki i składni, kliknij tutaj. Warto również pamiętać, że w tekście VEXcode V5 dostępna jest pomoc.

Programming Drive Forward and Reverse - VEXcode V5 Text

The Speedbot is ready to move!

This exploration will give you the tools to be able to program your Speedbot for simple movements. At the end of this activity, you will engage in the Basetball Drills Challenge using forward, reverse, and waiting behaviors.

- VEXcode V5 Text instructions that will be used in this exploration:
 - Drivetrain.driveFor(1, inches);
 - o wait(1, seconds);
- To find out more information about the instruction, select Help and then select Command Help. For more information on the help feature of VEXcode V5 Text click here.

• Make sure you have the hardware required, your engineering notebook, and VEXcode V5 Text downloaded and ready.

Hardware/Software Required:

Amount	Hardware/Other Items
1	Speedbot Robot
1	Charged Robot Battery
1	VEXcode V5 Text
1	USB Cable (if using a computer)
1	Engineering Notebook

1. Przygotowanie do analizy

Czy przed rozpoczęciem ćwiczenia masz gotowy każdy z tych elementów? Konstruktor powinien sprawdzić każdą z następujących kwestii:

- Czy wszystkie silniki i czujniki są podłączone do właściwego portu?
- Czy kable są poprawnie włożone do wszystkich silników i czujników?
- Czy mózg / jednostka sterująca jest włączony/a?
- Czy bateria jest naładowana?

2. Rozpocznij nowy projekt

Wykonaj następujące kroki, aby rozpocząć projekt:

• Otwórz menu Plik i wybierz Otwórz przykłady.

 Wybierz i otwórz szablon Speedbot (Drivetrain 2-motor, No Gyro). Szablon zawiera konfigurację silnika Speedbota. Jeśli nie jest używany, twój robot nie uruchomi projektu poprawnie.

Competition Template	Competition template with no devices configured
SDV (Drivetrain 2-motor)	Blank Pre-Configured SDV Drivetrain Project
SDV (Drivetrain 4-motor)	Blank Pre-Configured SDV 4-motor Drivetrain Project
SDV (Motors)	Blank Pre-Configured SDV Project
Speedbot (Drivetrain 2-motor, No Gyro)	Blank Pre-Configured V5 Speedbot 2-motor Drivetrain Project
Speedbot (Drivetrain 4-motor, No Gyro)	Blank Pre-Configured V5 Speedbot 4-motor Drivetrain Project
Speedbot (Motors)	Blank Pre-Configured V5 Speedbot Project
	Cancel Next

• Ponieważ będziesz pracować nad przemieszczaniem Speedbota do przodu i do tyłu, nadasz projektowi nazwę Drive.

Name: MyProject1		Name: Drive	
	Cancel Create		Cancel Create

• Po zakończeniu wybierz Utwórz.

xample	Project		
Name:	Drive		
		Cancel	Create

• Sprawdź, czy nazwa projektu **Drive** znajduje się teraz w oknie pośrodku paska narzędzi.

				1	2	Drive	
<	G.	main	.cpp				
	1	/*					
	2	/*					
	3	/*	Module:	n	nain.cpp		
	4	/*	Author:	١	/EX		

3. Jazda do przodu

Możesz teraz rozpocząć programowanie robota do jazdy do przodu!

 Zanim zaczniemy programować, musimy zrozumieć, czym jest instrukcja. Instrukcja składa się z trzech części. Więcej informacji na temat instrukcji można znaleźć w artykule "Zasady i wytyczne dotyczące składni - programowanie tekstowe VEXcode V5".

• Dodaj instrukcję do projektu:

 Wybierz ikonę Slot, aby wybrać jedno z ośmiu dostępnych slotów w Mózgu i wybierz slot 1.

• Podłącz Mózg do komputera za pomocą kabla micro USB i uruchom go. Ikona Mózg na pasku narzędzi zmienia kolor na zielony po pomyślnym nawiązaniu połączenia.

1	Drive	Ø	⇒ □	*	►			?	
				*/ */			54. E	ivi	
EXCODE CONFIG	GURED DEVICES						State of the second	81.77"	

 Gdy Mózg V5 est podłączony do komputera, ikona Buduj zmienia się na ikonę Pobierz. Wybierz Pobierz, aby pobrać projekt do mózgu.

🗊 🕎 Drive 😝 🗔 🚳		🗊 🖞 Drive 🥽 🗔 🛃 🕨	
*/		*/	□ 8. Ko
	ED. EXTENSION E. EXTENSION Margine and EXTENSION To a 1 and a second and a second and the second and the second and the second and the second and the second and the second and the second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the test second and the second and the second and the second and the test second and the second and the test second and the second and	*/	R. Billion and the second second

• Sprawdź, czy projekt został pobrany, patrząc na ekran Mózgu. Nazwa projektu Drive powinna być wymieniona w slocie 1.

• Uruchom projekt, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk **Uruchom** w mózgu robota. Gratulujemy stworzenia pierwszego projektu!

4. Jedź do tyłu

Teraz, gdy zaprogramowałeś robota do jazdy do przodu, zaprogramujmy go do jazdy do tyłu.

• Zmień parametr w instrukcji napędu na -100.

• Wybierz nazwę projektu, aby zmienić ją z Drive na Reverse.

	Drive	👔 💆 Reverse
	Project Name	Project Name
-	Drive	Reverse
	Description	Description
	Empty V5 C++ Project	gi
*	Enable Expert Robot Configuration	Enable Expert Robot Configuration
h	Enable Expert Autocomplete	h' Enable Expert Autocomplete
e		е .

• Wybierz ikonę Slot, aby wybrać nowy slot. Wybierz slot 2.

• Pobierz projekt.

• Sprawdź, czy projekt został pobrany, patrząc na ekran Mózgu. Nazwa projektu Reverse powinna być wymieniona w slocie 2.

Uruchom projekt, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk **Uruchom.**

5. Poczekaj, a następnie jedź wstecz

Teraz, gdy zaprogramowaliśmy robota do jazdy do przodu, a następnie do tyłu, możemy teraz dodać instrukcję oczekiwania, aby robot czekał przez określony czas przed jazdą do tyłu.

 Dodaj instrukcję, jak pokazano poniżej. Nakazuje ona robotowi odczekać trzy sekundy przed jazdą do tyłu.

• Wybierz nazwę projektu, aby zmienić ją z Reverse na WaitReverse.

	2 WaitReverse	
	Project Name	
gı	WaitReverse	ľ
	Description	
<(Empty V5 C++ Project	
1'		
0	Enable Expert Robot Configuration	
ir);	Enable Expert Autocomplete	!

• Wybierz ikonę Slot, aby wybrać nowy slot - 3ci.

• Pobierz projekt.

• Sprawdź, czy projekt został pobrany, patrząc na ekran Mózgu. Nazwa projektu WaitReverse powinna być wymieniona w slocie 3.

• Uruchom projekt, upewniając się, że jest wybrany, a następnie naciśnij przycisk **Uruchom.**

6. Ukończ wyzwanie Basketball Drills!

Wygląd boiska do wyzwania Basketball Drills

W wyzwaniu Basketball Drills robot musi być w stanie pokonywać szereg linii z różnych odległości. Robot przejedzie do przodu do pierwszej linii, która znajduje się 10 cm od pozycji wyjściowej, odczeka 1 sekundę, a następn ie cofnie się, powracając do tej samej linii. Robot następnie powtórzy czynność, podjeżdżając 20 cm do przodu do drugiej linii, odczeka 1 sekundę, a następnie pojedzie wstecz do pierwotnej linii startu. Robot podjedzie do trzeciej linii w odległości 40 cm, odczeka 1 sekundę, a następnie wróci na linię startu, aby zakończyć wyzwanie.

Przed zaprogramowaniem robota zaplanuj ścieżkę i zachowanie robota w notatniku inżyniera.

Po ukończeniu wyzwania będziesz mógł łączyć ruchy do przodu i do tyłu z dodatkowymi zachowaniami robota, aby ukończyć jeszcze bardziej zaawansowane wyzwania.

Podczas programowania należy pamiętać, że 1 cm = 10 mm

Programowanie skrętu w prawo i lewo – programowanie tekstowe

Speedbot jest gotowy do skręcania!

Te zajęcia dostarczą Ci narzędzi, dzięki którym będziesz mógł rozpocząć tworzenie ciekawych projektów dla Twojego Speedbota.

• Instrukcje jakie zostaną użyte:

Drivetrain.turnFor(90, degrees);

 Aby uzyskać więcej informacji na temat instrukcji, wybierz Pomoc, a następnie wybierz Pomoc do instrukcji. Aby uzyskać więcej informacji na temat funkcji pomocy VEXcode V5 Text, kliknij tutaj.

 Upewnij się, że masz wymagany sprzęt, notes inżyniera i program VEXcode V5 pobrany i gotowy do użycia.

Sprzęt/ oprogramowanie:

llość	Sprzęt/inne przedmioty
1	Speedbot
1	Naładowana bateria robota
1	Program VEXcode V5 Text
1	Kabel USB (jeśli używasz komputera)
1	notes inżyniera

1. Przygotowanie do analizy

Czy przed rozpoczęciem ćwiczenia masz gotowy każdy z tych elementów?

- Czy silniki są podłączone do odpowiednich portów?
- Czy kable są poprawnie włożone do wszystkich silników i czujników?
- Czy mózg / jednostka sterująca jest włączony/a?
- Czy bateria jest naładowana?

2. Rozpocznij nowy projekt

Wykonaj następujące kroki, aby rozpocząć projekt:

• Otwórz menu Plik i wybierz Otwórz przykłady.

• Wybierz i otwórz szablon Speedbot (Drivetrain 2-motor, No Gyro). Szablon zawiera konfigurację silnika Speedbota. Jeśli szablon nie jest używany, twój robot nie uruchomi projektu poprawnie.

Exan	nples	
	Competition Template	Competition template with no devices configured
	SDV (Drivetrain 2-motor)	Blank Pre-Configured SDV Drivetrain Project
	SDV (Drivetrain 4-motor)	Blank Pre-Configured SDV 4-motor Drivetrain Project
	SDV (Motors)	Blank Pre-Configured SDV Project
	Speedbot (Drivetrain 2-motor, No Gyro)	Blank Pre-Configured V5 Speedbot 2-motor Drivetrain Project
	Speedbot (Drivetrain 4-motor, No Gyro)	Blank Pre-Configured V5 Speedbot 4-motor Drivetrain Project
	Speedbot (Motors)	Blank Pre-Configured V5 Speedbot Project
		Cancel Next

 Ponieważ będziesz pracować nad obracaniem Speedbota, nadasz projektowi nazwę Turn.

Example Project			
Name:	Turn		
		Cancel	Create

• Po zakończeniu wybierz Utwórz.

Example	Project				
Name:	Turn				
			Ca	ncel	Create

3. Skręć w prawo

Możesz teraz rozpocząć programowanie robota, aby skręcił w prawo!

• Dodaj instrukcję do projektu:

20	<pre>#include "vex.h"</pre>
21	
22	using namespace vex;
23	
24	<pre>int main() {</pre>
25	<pre>// Initializing Robot Configuration. D0 NOT REMOVE!</pre>
26	<pre>vexcodeInit();</pre>
27	
28	<pre>Drivetrain.turnFor(90, degrees);</pre>
29	}

- Wybierz ikonę Slot, aby wybrać jedno z ośmiu dostępnych miejsc w Mózgu i wybierz slot 1.
- Podłącz Mózg do komputera za pomocą kabla micro USB i włącz go. Ikona Mózg na pasku narzędzi zmienia kolor na zielony po pomyślnym nawiązaniu połączenia.
- Gdy V5 jest podłączony do komputera, ikona Buduj zmienia się na ikonę Pobierz.
 Wybierz **Pobierz**, aby pobrać projekt do mózgu.
- Sprawdź, czy projekt został pobrany, patrząc na ekran Mózgu. Nazwa projektu Drive powinna być wymieniona w slocie 1.

 Uruchom projekt na robocie, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk Uruchom w Mózgu robota.

4. Skręć w lewo

Teraz, gdy zaprogramowałeś swojego robota, aby skręcał w prawo, pozwól nam zaprogramować go, aby skręcał w lewo.

• Zmień parametr w programie, aby wyświetlał -90 zamiast 90.

- Wybierz nazwę projektu, aby zmienić ją z Turn na TurnLeft.
- Wybierz ikonę Slot, aby wybrać nowy slot. Wybierz slot nr 2.
- Pobierz projekt.
- Sprawdź, czy projekt został pobrany, patrząc na ekran Mózgu robota. Nazwa projektu TurnLeft powinna być wymieniona w slocie 2.
- Uruchom projekt na robocie, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk Uruchom.

5. Poczekaj, a następnie skręć w lewo

Teraz, gdy zaprogramowaliśmy robota, aby skręcał w lewo, w prawo, a następnie znów w lewo, możemy dodać instrukcję czekania, aby robot czekał przez określony czas przed skręceniem w lewo.

• Dodaj instrukcję *czekaj.* To nakazuje robotowi odczekać trzy sekundy przed skręceniem.

- Wybierz nazwę projektu, aby zmienić ją z TurnLeft na WaitLeft.
- Wybierz ikonę Slot, aby wybrać nowy slot. Wybierz slot nr 3.
- Pobierz projekt.
- Sprawdź, czy projekt został pobrany, patrząc na ekran Mózgu. Nazwa projektu WaitLeft powinna być wymieniona w slocie 3.
- Uruchom projekt na robocie, upewniając się, że projekt jest wybrany, a następnie naciśnij przycisk Uruchom.

6. Ukończ wyzwanie Kamery Bezpieczeństwa!

Wyzwanie kamery bezpieczeństwa

W wyzwaniu kamery bezpieczeństwa robot musi zacząć od skrętu w prawo o 90 stopni. Następnie obróci się w lewo o 180 stopni, a później w prawo o 180 stopni "skanując", tak jak kamera bezpieczeństwa. Następnie robot powtórzy obrót o 180 stopni w lewo i 180 stopni w prawo jeszcze dwa razy. W ten sposób robot przeskanuje łącznie zakres 180 stopni trzy razy.

Przed zaprogramowaniem robota zaplanuj ścieżkę i zachowanie robota w notatniku inżyniera.

Po ukończeniu wyzwania będziesz mógł łączyć ruchy do przodu i do tyłu z dodatkowymi zachowaniami robota, aby ukończyć jeszcze bardziej zaawansowane wyzwania.

Rozwiąż inne problemy XXI wieku, stosując podstawowe umiejętności i koncepcje, których się nauczyłeś.

Roboty w medycynie

Robot kompletujący receptę pacjenta w aptece

Roboty zwiększają wydajność szpitali

Stosowanie technologii wpływa na ludzi w różny sposób, w tym na ich bezpieczeństwo i wygodę. Medycyna w większym stopniu polega na robotach, które wykonują powtarzalne, codzienne zadania w szpitalach. Ponieważ szpitale mają tendencję do przemieszczania wielu materiałów w budynkach w ciągu dnia, istnieje duże zapotrzebowanie na roboty do wykonywania zadań związanych z dostawami i transportem. Wykonywanie tego rodzaju zadań przez maszyny zwiększa wydajność personelu szpitala, oszczędza pieniądze i pozwala pielęgniarkom i lekarzom skupić się na opiece nad pacjentem.

Jednym z materiałów, które roboty mogą bezpiecznie i szybko dostarczyć, sa leki. Gdy farmaceuci wprowadzają recepty do swoich komputerów, roboty wybierają właściwy rodzaj i dawkę, skanując prawidłowe kody kreskowe. Robot następnie komplatuje i znakuje leki, śledząc, czy właściwy lek dotrze do potrzebującego pacjenta. Te roboty dostawcze mogą przenosić oznakowane leki do stacji pielęgniarskich, a nawet pokojów poszczególnych pacjentów. Jest to bardzj skuteczna metoda, która może przyspieszyć dostarczanie pacjentom ważnych leków, pomóc wypełnić braki kadrowe i przechowywać recepty w bezpiecznym miejscu podczas transportu.

Niektóre roboty medyczne będą podróżować ponad 400 mil w ciągu tygodnia, wykonując swoje obchody, poruszając się po korytarzach, jeżdżąc windami i zatrzymując się na różnych stacjach. Oprócz dostarczania materiałów medycznych, niektóre roboty odwiedzają nawet pacjentów, aby sprawdzić ich stan i podać parametry życiowe lekarzom. Wraz z rozwojem robotyki pacjenci będą nadal odnosić coraz więcej korzyści z obecności robotów w środowisku szpitalnym.

Planowanie projektów w zawodach VEX

Dwóch zawodników VEX przygotowujących się do wyzwania programistycznego

Tworzenie zachowań rutynowych i powtarzalnych

Zawody VEX Robotics wymagają zaplanowania autonomicznej rutyny dla 60-sekundowego meczu umiejętności programowania w ramach wyzwania Robot Skills Challenge oraz dla 15-sekundowego okresu autonomicznego w tegorocznej grze. Programowanie robota do autonomicznej pracy wymaga rozbicia tego, co robot musi zrobić, na programowalny kod oparty na jego zachowaniu.

Oto kilka typowych zachowań robota konkursowego VEX:

- Jazda do przodu i do tyłu
- Skręcanie w prawo i w lewo
- Chwytanie obiektu gry
- Precyzyjne umieszczenie przedmiotu gry w danym miejscu
- Sortowanie różnych obiektów w grze

Kiedy już stworzysz podstawowe instrukcje dla tego typu zachowań, przeprowadzenie udanej autonomicznej procedury oznacza po prostu zaplanowanie kolejności tych działań!

Czy istnieje skuteczniejszy sposób, aby dojść do tego samego wniosku? zastanów się nad tym czego się nauczyłeś i spróbuj to ulepszyć.

Przygotuj się na wyzwanie Automed

Układ szpitala w wyzwaniu Automed z wymiarami

Przygotuj się do wyzwania Automed

W tym wyzwaniu musisz zaprogramować robota do poruszania się po szpitalu, ponieważ dostarcza on leki pacjentom w kilku różnych pokojach. Aby pomyślnie ukończyć to wyzwanie, musisz zaprogramować swojego robota, aby przemieszczał się ze strefy startowej i zatrzymywał się w aptece na 5 sekund w celu pobrania leków.

Projekt powinien następnie przesunąć robota do każdej z sal pacjentów (1, 2 i 3 w dowolnej kolejności) i zatrzymać się w pokoju na 3 sekundy, aby podać lek.

Aby poruszać się między piętrami w szpitalu, robot musi zatrzymać się w obszarze oznaczonym jako winda przez 5 sekund, aby wskazać, że przemieszcza się na inne piętro.

Po dostarczeniu leków do wszystkich pomieszczeń robot powinien powrócić do Strefy Startowej. Powinieneś odtworzyć powyższą mapę szpitala na swojej podłodze za pomocą taśmy i miarki lub linijki. Pamiętaj, aby wskazać, gdzie znajduje się strefa startowa, apteka, winda i pokoje dla pacjentów.

Aby ukończyć wyzwanie, będziesz potrzebować:

- Rolka taśmy (aby stworzyć układ szpitala na podłodze)
- Otwarta przestrzeń o wymiarach 1,8 m na 2 m
- Miernik lub linijka

Projektuj, rozwijaj i iteruj swój projekt - bloki VEXcode V5

Podczas projektowania rozwiązania odpowiedz na następujące pytania w swoim notatniku inżyniera:

- Jakie zadania robota chcesz zaprogramować? Wyjaśnij szczegółowo.
- Jakie kroki podejmiesz, aby przetestować projekt? Wyjaśnij szczegółowo.
- W jaki sposób można zaprogramować robota, aby wydajniej wykonywał zadanie? Wyjaśnij jak.

Podczas tworzenia projektu wykonaj poniższe czynności:

- Za pomocą linijki zaplanuj ścieżkę, którą chcesz aby robot pokonał, użyj rysunków i pseudokodu. Oceń swój pseudokod przed przejściem do drugiego kroku.
- Użyj pseudokodu, który stworzyłeś, aby rozwinąć swój projekt.
- Często testuj swój projekt i powtarzaj go, korzystając z tego, czego nauczyłeś się podczas testów.

Jeśli masz problemy z rozpoczęciem, przejrzyj przykładowe projekty.

Przeglądanie przykładowych projektów może pomóc w generowaniu pomysłów.

Wykonaj następujące kroki:

- Otwórz menu Plik.
- Wybierz **Otwórz** przykłady.

Wyzwanie Automed – bloki VEXcode V5

Układ szpitala w wyzwaniu Automed z wymiarami

Wyzwanie Automed

W tym wyzwaniu musisz zaprogramować robota do poruszania się po szpitalu, ponieważ dostarcza on leki pacjentom w kilku różnych pokojach.

Zasady:

- Robot musi zaczynać i kończyć w strefie startowej.
- Cały robot musi znajdować się w aptece, windzie i pokojach pacjentów i odczekać następujące czasy, aby wykonać czynności:
- Apteka: Odczekaj co najmniej 5 sekund, aby odebrać leki.
- Winda: odczekaj co najmniej 5 sekund, aby dotrzeć na kolejne piętro.
- Pokój dla pacjentów: odczekaj co najmniej 3 sekundy, aby zostawić leki.
- Robot nie może stykać się ani przechodzić przez żadne ściany.
- Robot musi najpierw udać się do apteki, aby odebrać leki dla pacjentów.
- Robot musi odwiedzić każdą z sal pacjentów (w dowolnej kolejności), aby zostawić leki.
- Baw się dobrze!
Projektuj, rozwijaj i iteruj swój projekt – programowanie tekstowe

Podczas projektowania rozwiązania odpowiedz na następujące pytania w swoim notatniku inżyniera:

- Jakie zadania robota chcesz zaprogramować? Wyjaśnij szczegółowo.
- Jakie kroki podejmiesz, aby przetestować projekt? Wyjaśnij szczegółowo.
- W jaki sposób można zaprogramować robota, aby wydajniej wykonywał zadanie? Wyjaśnij jak.

Podczas tworzenia projektu wykonaj poniższe czynności:

- Za pomocą linijki zaplanuj ścieżkę, którą chcesz aby robot pokonał, użyj rysunków i pseudokodu. Oceń swój pseudokod przed przejściem do drugiego kroku.
- Użyj pseudokodu, który stworzyłeś, aby rozwinąć swój projekt.
- Często testuj swój projekt i powtarzaj go, korzystając z tego, czego nauczyłeś się podczas testów.

Jeśli masz problemy z rozpoczęciem, przejrzyj przykładowe projekty.

Przeglądanie przykładowych projektów może pomóc w generowaniu pomysłów.

\ 5	File Edit Tools	
	New ¥N	
	New Window	
Inclu	Open #0	
▶ src	Open Recent >	
	Open Examples	
	Open Tutorials	ted
	Import	rip
	Export	

Wykonaj następujące kroki:

- Otwórz menu Plik.
- Wybierz Otwórz przykłady.

Exan	Examples		
	Competition Template	Competition template with no devices configured	
	SDV (Drivetrain 2-motor)	Blank Pre-Configured SDV Drivetrain Project	
	SDV (Drivetrain 4-motor)	Blank Pre-Configured SDV 4-motor Drivetrain Project	
	SDV (Motors)	Blank Pre-Configured SDV Project	
	Speedbot (Drivetrain 2-motor, No Gyro)	Blank Pre-Configured V5 Speedbot 2-motor Drivetrain Project	
	Speedbot (Drivetrain 4-motor, No Gyro)	Blank Pre-Configured V5 Speedbot 4-motor Drivetrain Project	
	Speedbot (Motors)	Blank Pre-Configured V5 Speedbot Project	
		Cancel Next	

Wyzwanie Automed – programowanie tekstowe



Układ szpitala w wyzwaniu Automed z wymiarami

Wyzwanie Automed

W tym wyzwaniu musisz zaprogramować robota do poruszania się po szpitalu, ponieważ dostarcza on leki pacjentom w kilku różnych pokojach.

Zasady

- Robot musi zaczynać i kończyć w strefie startowej.
- Cały robot musi znajdować się w aptece, windzie i pokojach pacjentów i odczekać następujące czasy, aby wykonać czynności:
- Apteka: Odczekaj co najmniej 5 sekund, aby odebrać leki.
- Winda: odczekaj co najmniej 5 sekund, aby dotrzeć na kolejne piętro.
- Pokój dla pacjentów: odczekaj co najmniej 3 sekundy, aby zostawić leki.
- Robot nie może stykać się ani przechodzić przez żadne ściany.
- Robot musi najpierw udać się do apteki, aby odebrać leki dla pacjentów.
- Robot musi odwiedzić każdą z sal pacjentów (w dowolnej kolejności), aby zostawić leki.
- Baw się dobrze!



Zrozum podstawowe pojęcia i dowiedz się, jak zastosować je w różnych sytuacjach. Ten proces powtórki będzie motywował do nauki.

Powtórzenie – bloki VEXcode V5

1. W programowaniu robotów, co nazywa się "zachowaniem"?

- o Złożone zadanie
- Sposób działania robota
- Proste zadanie
- Wszystkie te odpowiedzi są poprawne.
- 2. Prawda czy fałsz: rozkładanie złożonych zachowań robota na mniejsze, proste kroki nazywa się dekompozycją.
 - o Prawda
 - o Fałsz
- 3. Prawda czy fałsz: aby program mógł zostać uruchomiony, należy go pobrać do Slotu 1.
 - o Prawda
 - o Fałsz
- 4. Prawda czy fałsz: Musisz wybrać szablon za każdym razem, gdy rozpoczynasz nowy projekt.
 - o Prawda
 - o Fałsz
- 5. O ile stopni musi obrócić się robot, aby spojrzeć w przeciwnym kierunku?
 - o 180 stopni
 - o 90 stopni
 - o 45 stopni
 - o 360 stopni
- 6. Prawda czy fałsz: pseudokod to rzeczywisty język programowania, który Twój robot może zrozumieć.
 - o Prawda
 - o Fałsz

Powtórzenie – programowanie tekstowe

7. W programowaniu robotów, co nazywa się "zachowaniem"?

- Złożone zadanie
- o Sposób działania robota
- Proste zadanie
- o Wszystkie te odpowiedzi są poprawne.

8. Prawda czy fałsz: Polecenie *driveFor* może służyć do kierowania robota do przodu na określoną odległość.

- o Prawda
- o Fałsz
- 9. Prawda czy fałsz: Polecenie wait zakończy projekt.
 - o Prawda
 - o Fałsz

10. Prawda czy fałsz: Polecenie *driveFor* może służyć do poruszania robota do przodu i do tyłu.

- o Prawda
- o **Fałsz**

11. Aby obrócić robota o 180 stopni:

- Użyj polecenia *turnFor* i zmień stopnie na 180.
- Użyj polecenia *driveFor* i ustaw odległość na 180.
- Użyj polecenia *driveFor* i zmień stopnie z 180 na -180, aby skręcić do tyłu.
- Użyj polecenia *wait* i ustaw sekundy na 180.

12. Prawda czy fałsz: pseudokod to rzeczywisty język programowania, który Twój robot może zrozumieć.

- Prawda
- Fałsz
- 13. Która z poniższych serii instrukcji przełożyłaby następujący pseudokod na zachowania?



- o driveFor (forward), turnFor (left), wait, turnFor (left), driveFor (reverse)
- turnFor (right), wait, driveFor (forward)
- o driveFor (forward), turnFor (right), wait, turnFor (right), driveFor (forward)
- o driveFor (forward), turnFor (right), turnFor (right), driveFor (forward)

APPENDIX

Dodatkowe informacje, zasoby i materiały.

Używanie 1 Post Hex Nut Retainer w/ Bearing Flat



1 Post Hex Nut Retainer w/ Bearing Flat

Używanie 1 Post Hex Nut Retainer w/ Bearing Flat

1 Post Hex Nut Retainer w/ Bearing Flat umożliwia płynne obracanie się wałów przez otwory w elementach konstrukcyjnych. Po zamontowaniu zapewnia dwa punkty styku na elementach konstrukcyjnych w celu zapewnienia stabilności. Na jednym końcu elementu znajduje się słupek o wymiarach umożliwiających bezpieczne dopasowanie do kwadratowego otworu elementu konstrukcyjnego. Środkowy otwór elementu ma taki rozmiar i szczelinę, aby bezpiecznie dopasować nakrętkę sześciokątną, umożliwiając łatwe dokręcenie śruby 8-32 bez użycia klucza lub kombinerek. Otwór na końcu elementu jest przeznaczony do przechodzenia wałków lub śrub.

Aby skorzystać z retainera:

 Wyrównaj go na elemencie konstrukcyjnym VEX tak, aby otwór końcowy znajdował się w żądanym miejscu, a sekcja środkowa i końcowa również były podparte przez element konstrukcyjny.

- Włóż kwadratowy słupek wystający z elementu do elementu konstrukcyjnego, aby pomóc go utrzymać na miejscu.
- Włożyć nakrętkę sześciokątną w środkową część retainera tak aby zrównała się z resztą elementu.
- W stosownych przypadkach wyrównaj wszelkie dodatkowe elementy konstrukcyjne z tyłu głównego elementu konstrukcyjnego.
- Użyj śruby 8-32 odpowiedniej długości, aby przymocować elementy konstrukcyjne do retainera przez środkowy otwór i nakrętkę sześciokątną.

Używanie 4 Post Hex Nut Retainer



4 Post Hex Nut Retainer

Używanie 4 Post Hex Nut Retainer

4 Post Hex Nut Retainer zapewnia pięć punktów styku do tworzenia wytrzymałego połączenia między dwoma elementami konstrukcyjnymi za pomocą jednej śruby i nakrętki. Każdy narożnik retainera zawiera słupek o rozmiarze umożliwiającym bezpieczne dopasowanie go do kwadratowego otworu w elemencie konstrukcyjnym. Środek elementu ma taki rozmiar i szczelinę, aby bezpiecznie dopasować nakrętkę sześciokątną, umożliwiając łatwe dokręcenie śruby 8-32 bez użycia klucza lub kombinerek.

Aby skorzystać z retainera:

- Wyrównaj go na elemencie konstrukcyjnym VEX tak, aby środkowy otwór znajdował się w żądanym miejscu, a każdy narożnik był wsparty elementem konstrukcyjnym.
- Włóż kwadratowy słupek wystający z elementu do elementu konstrukcyjnego, aby pomóc go utrzymać na miejscu.
- Włożyć nakrętkę sześciokątną w środkową część retainera tak aby zrównała się z resztą elementu.
- W stosownych przypadkach wyrównaj wszelkie dodatkowe elementy konstrukcyjne z tyłu głównego elementu konstrukcyjnego.
- Użyj śruby 8-32 odpowiedniej długości, aby przymocować elementy konstrukcyjne do retainera przez środkowy otwór i nakrętkę sześciokątną.

Używanie 1 Post Hex Nut Retainer



1 Post Hex Nut Retainer

Używanie 1 Post Hex Nut Retainer

1 Post Hex Nut Retainer zapewnia dwa punkty styku do łączenia elementu konstrukcyjnego z innym elementem za pomocą jednej śruby i nakrętki. Jeden koniec elementu zawiera słupek o wymiarach umożliwiających bezpieczne dopasowanie do kwadratowego otworu elementu konstrukcyjnego. Drugi koniec retainera ma taki rozmiar i szczelinę, aby bezpiecznie dopasować nakrętkę sześciokątną, umożliwiając łatwe dokręcenie śruby 8-32 bez użycia klucza lub kombinerek.

Aby skorzystać z retainera:

- Wyrównaj go na elemencie konstrukcyjnym VEX tak, aby oba końce były podparte przez element konstrukcyjny i ustawione tak, aby zamocować drugi element.
- Włóż kwadratowy słupek wystający z retainera do elementu konstrukcyjnego, aby pomóc go utrzymać na miejscu.
- Jeśli element jest używany do mocowania dwóch elementów konstrukcyjnych, włóż nakrętkę sześciokątną do drugiego końca retainera, tak aby był wyrównany z resztą elementu. W przypadku użycia do zamocowania innego rodzaju elementu, takiego jak np.: element dystansowy, może być właściwe włożenie śruby przez tę stronę.

- Jeśli jest to niezbędne, wyrównaj wszelkie dodatkowe komponenty z tyłu głównego elementu konstrukcyjnego.
- Jeśli retainer jest używany do łączenia dwóch elementów konstrukcyjnych, użyj śruby 8-32 o odpowiedniej długości, aby zabezpieczyć elementy konstrukcyjne przez otwór i nakrętkę sześciokątną. Jeśli jest używany do łączenia innego typu elementu, takiego jak np.: element dystansowy, zabezpiecz go bezpośrednio lub za pomocą nakrętki sześciokątnej.

Notes inżyniera

march 1074 1876 see you To my delight he came and declared That he and heard and understand what I said , rig 1. MD I asked him to repeat the words - the mint He areneved "Jon said " M. Wateou - come here. I want to see you"." We then changed places and I listened at 5 while M. Watson read a few passages from a book into the month piece M. It was cutainly the case That articulate sounds proceeded from S. The effect was load but indictinet and muffled : 1. The improved instrument shower in Fig. I was If I had read beforehand The passage given by Mr. Water I should have recognized constructed this morning and tried this latting . Pis a trass pipe and W The platimum wire every word. As it was I could not M the month piece and S The armetine of The Keceiving Instrument . word here and there was quite distinct. I made cathe "to" and "out" and "further"; W. Watson was stationed in one room with the Receiving fistorment . He pressed one " ear closely against S and closell his other and finally the sentence " Mr. Bell Do your ear with his hand . The Transmitting Instrument understand what I day? Do- you - un der - stand - what - I - Say " came was placed in another room and the doors of both rooms were closed. was andible when The armature S was re-I then shouted into M the following sentence: "W" Watson - Come here - I want to neoned .

Notatki Alexandra Grahama Bell'a z udanego eksperymentu z jego pierwszym telefonem

Notatnik inżyniera dokumentuje Twoją pracę

Nie tylko używasz notatnika do organizowania i dokumentowania swojej pracy, ale jest to także miejscem do refleksji nad działaniami i projektami. Podczas pracy w zespole każdy członek zespołu będzie prowadził własny dziennik, aby ułatwić współpracę w grupie.

Twój notatnik inżyniera powinien mieć następujące elementy:

- Wpis na każdy dzień lub sesję, w której pracowałeś nad rozwiązaniem
- Wpisy chronologiczne, z datą każdego wpisu
- Jasne, schludne i zwięzłe notatki, dobrze zorganizowane
- Etykiety, aby czytelnik zrozumiał wszystkie Twoje notatki i ich dopasowanie do iteracyjnego procesu projektowania

Wpis może obejmować:

- Burzę mózgów
- Szkice lub zdjęcia prototypów
- Pseudokod i schematy blokowe planowania
- Wszelkie zastosowane obliczenia lub algorytmy Odpowiedzi na pytania przewodnie
- Uwagi dotyczące obserwacji i / lub przeprowadzonych testów
- Notatki i refleksje na temat różnych iteracji