

Modul 1

**ROBOTIKA PROSTREDNÍCTVOM POSUNKOVÉHO JAZYKA:
ZABEZPEČENIE PRÍSTUPU A ZAPOJENIA ŽIAKOV
SO ZDRAVOTNÝM POSTIHNUTÍM (NEPOČUJÚCICH ALEBO
NEDOSLÝCHAVÝCH) DO DIGITÁLNEHO SVETA KÓDOVANIA
A ROBOTIKY – ROBOTICS4DEAF**

Projekt č.: 2019-1-PL01-KA201-065123



Erasmus+

Tento projekt bol financovaný s podporou Európskej komisie.
Toto oznámenie vyjadruje len názory autora a Komisia
nenesie zodpovednosť za akékoľvek použitie
informácií v ňom obsiahnutých.



Ako čítať tento dokument

Náš vzdelávací materiál sa skladá z dvoch modulov, toto je konceptová verzia modulu 1. Pre lepšiu predstavu o štruktúre vzdelávacieho materiálu si pozrite kompetenčný rámec: (https://drive.google.com/file/d/1858gQJK052zVsk8ivsI5YEEHq2pNkFW_/view?usp=sharing)
-link k sk prekladom kompetenčného rámca

Modul 1 sa skladá z 2 častí. Každá časť sa skladá z niekoľkých kapitol a podkapitol. Ako vidíte v kompetenčnom rámci, máme dve verzie modulu 1, jednu pre každú vekovú skupinu.

Toto je „plná“ verzia určená pre deti od 15 do 21 rokov.

Okrem výroby vzdelávacieho materiálu v písomnej forme sme sa usilovali (pomocou pokynov v príručke) prispôbiť ho cieľovej skupine nášho projektu.

Majte na pamäti, že ide iba o návrhy; nie sme odborníci v tejto oblasti. Podľa nášho názoru je nemožné pripraviť vzdelávací materiál bez zohľadnenia študentov, ktorých oslovujete.

Inštrukcie: (inými slovami, čo znamenajú všetky tie farby?)

Farba **zelená**: táto časť bude preložená do posunkového jazyka.

Color **modrá**: táto časť bude videom.

Žiadna farba: jednoduchý text.

Pre lepšiu predstavu sme pripravili aj beta verziu jednej podkapitoly. Je tiež pripojený a nazýva sa „beta slučky“.

Čakáme na vašu spätnú väzbu. V prípade akýchkoľvek otázok nás neváhajte kontaktovať.

Obsah

1.	SMARC – SMARt Robotic and Coding.....	3
I.	Stavanie SMARC.....	3
II.	Naučte sa, ako urobiť SMARC pohybom.....	3
A.	Zoskupenie blokov.....	4
B.	Ako urobiť pohyb SMARCu.	6
C.	Ukážkové programy.....	7
III.	Koncept kódovania slučky.....	11
2.	Používanie senzorov so Smarcom	14
I.	Senzory.....	14
A.	<i>Lego Boost senzor farieb a vzdialenosti</i>	14
B.	Detekcia objektov	15
C.	Zisťovanie farieb	16
D.	If/Else bloky:	20
II.	Detekcia zvuku pomocou protokolu Smarc.....	21
A.	Reakcia na zvuk	21
B.	Bloky zvukového senzora:	21
III.	Navigácia v Smarc pomocou diaľkového ovládača	25
A.	Diaľkové ovládanie	25
B.	Bloky diaľkového ovládania:.....	26

1. SMARC – SMARt Robotic and Coding

Cieľom tejto časti je oboznámiť študentov so základnými pohybmi robotov. Študenti si zostavia robota SMARC a naučia sa ho programovať tak, aby vykonával základné pohyby. Naučia sa tiež, ako ho pomocou slučiek naprogramovať na opakované pohyby.

I. Stavanie SMARC

Podrobný sprievodca vytvorením protokolu SMARC (Emphasys).



SMARC Building
Instructions.pdf

II. Naučte sa, ako urobiť SMARC pohybom.

Na fotografii nižšie je zobrazené programovacie prostredie aplikácie.

Ukazuje tiež oblasť, kde sa nachádzajú programovacie bloky. Nájdete tam všetko, čo potrebujete na vytvorenie vlastných programov.

Strávte nejaký čas prehliadaním aplikácie.



Foto 1- Bloková paleta

A. Zoskupenie blokov.

V tejto časti sa dozviете, ako sú bloky zoskupené podľa ich farby. Poznanie všeobecných kategórií uľahčuje hľadanie správneho bloku podľa toho, čo chceme robiť. V nasledujúcich častiach podrobne preskúmame každú kategóriu.

- Pomocou **Yellow (žltých)** blokov môžete riadiť priebeh programu

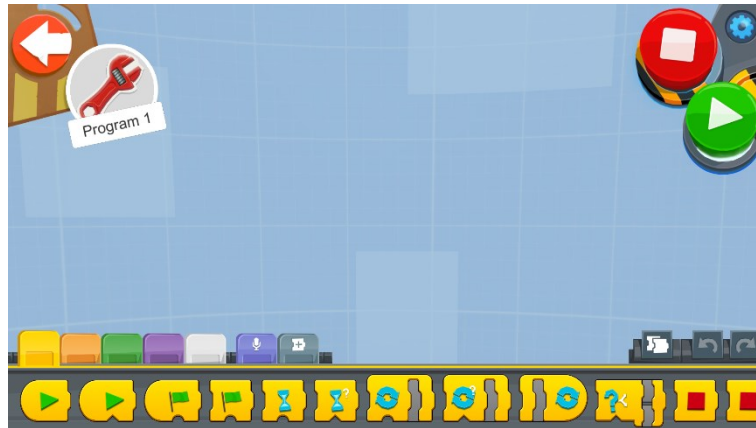


Foto 2-Žlté bloky

Môžete ich použiť na spustenie programu, zastavenie programu, pozastavenie programu alebo dokonca vytvorenie cyklu skupiny príkazov

- **Orange (*oranžové*)** bloky pracujú so Color and Distance Sensor (snímačom farieb a vzdialenosti), mikrofónom a Move Hub's Tilt Sensor (snímačom náklonu Move Hub)

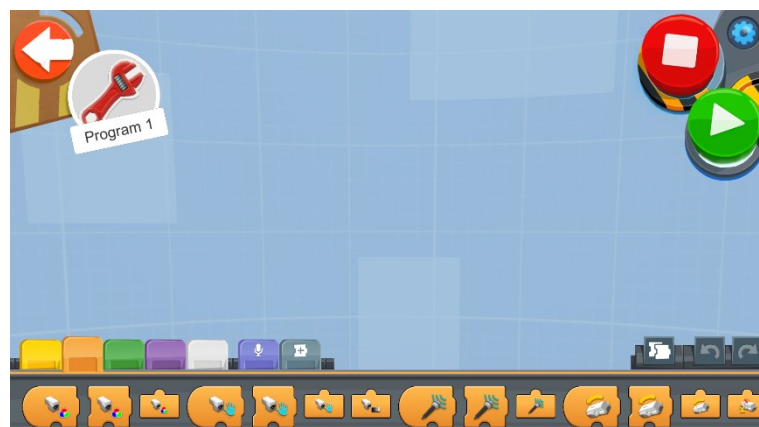


Foto 3-Oranžové bloky

Keď je aktivovaný jeden z vyššie uvedených senzorov, aktivuje sa tiež tieto bloky.

- **Purple (fialové)** bloky sa používajú na komunikáciu robota s vonkajším svetom.



Foto 4-Fialové bloky

Môžu prehrávať zvuky cez reproduktory zariadenia používateľa. Môžu tiež meniť farbu svetiel na zariadeniach Move Hub & Color and Distance Sensor (Move Hub a senzor farieb a vzdialenosti)

- **Biele** bloky vám umožňujú zložité programovanie pomocou “premenných” a “konštánt”.

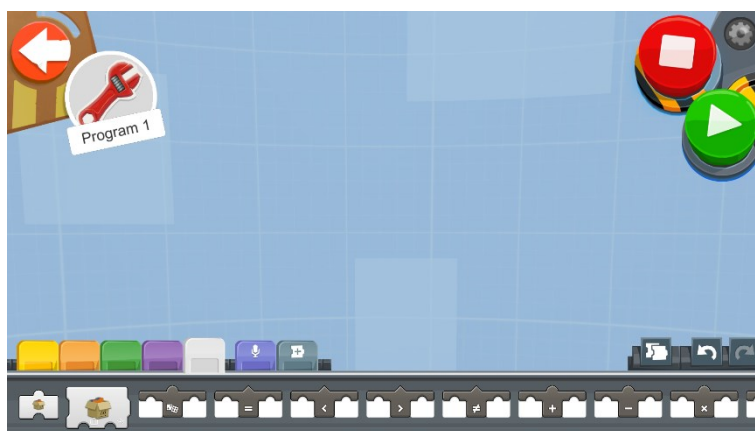


Foto 5-Biele bloky

Môžu byť tiež použité pri vykonávaní rôznych matematických výpočtov, vytváraní logických výrazov a generovaní náhodných čísel

- **Light Purple (svetlofialové)** bloky vám umožňujú zaznamenať zvuk, ktorý má robot opakovať.

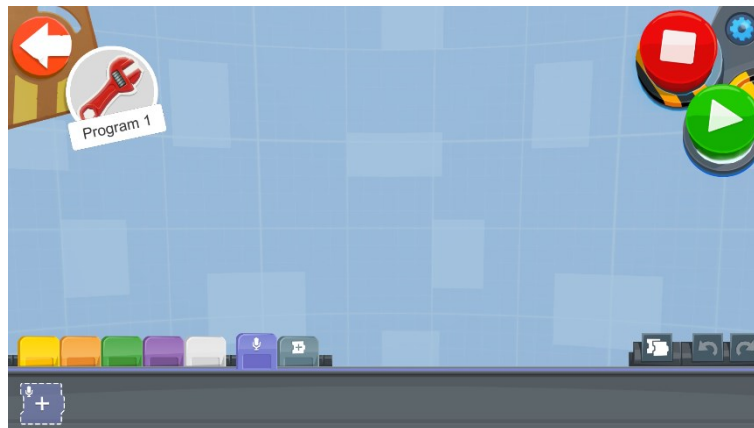


Foto 6-Svetlofialové bloky

- **Group Actions** block (blok **skupinových akcií**) vám umožňuje zoskupiť ako jedno tlačidlo bloku niekoľko akcií

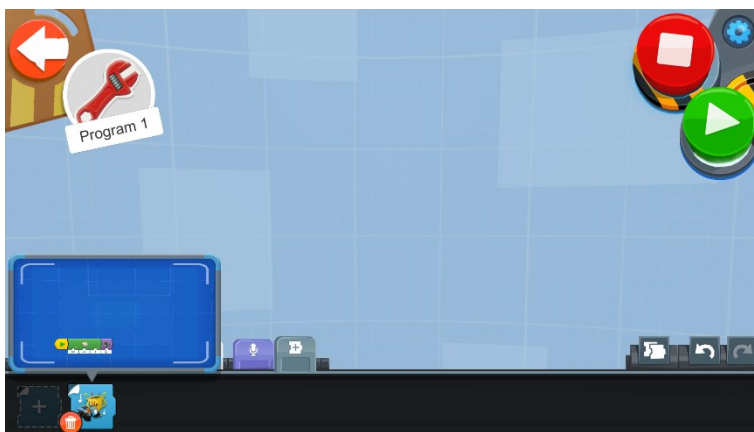


Foto 7- Bloky skupinových akcií

Toto je ekvivalent vytvárania podprogramov

B. Ako urobiť pohyb SMARCu.

Aby sme mohli začať programovať SMARC, prvá vec, ktorú potrebujeme, je StartBlock (štartovací blok) umiestnený v žltej palete



Existuje niekoľko blokov, ktoré riadia pohyb zariadenia Move Hub. Tieto bloky sa nachádzajú v zelenej sekcii palety. Tu uvádzame tie najbežnejšie.



Move Base for Duration (*presuňte základňu na dobu trvania*) - Motor s jednou rýchlosťou (od -100 do 100) a motor s dvoma rýchlosťami (od -100 do 100) po dobu trvania (v sekundách)



Move Base Tank for Distance (*posuňte základňu nádrže na vzdialenosť*) - motor s jednou rýchlosťou (od -100 do 100) a motor s dvoma rýchlosťami (od -100 do 100) pre vzdialenosť (v stupňoch)



Move Tank and Steer for Duration (*pohybujte nádržou a riadením po dobu trvania*) - otáčky oboch motorov (od -100 do 100) a smer riadenia (od -100 do 100) po dobu trvania (v sekundách)



Move Tank and Steer for Distance (*posuňte nádrž a riadenie na vzdialenosť*) - rýchlosť oboch motorov (od -100 do 100) a smer riadenia (od -100 do 100) na vzdialenosť (v stupňoch)



Move Tank (*pohybujte nádrž*) - motor s jednou rýchlosťou (od -100 do 100) a motor s dvoma rýchlosťami (od -100 do 100)



Move Tank and Steer for Duration (*pohybujte nádržou a riadením po dobu trvania*) - otáčky oboch motorov (od -100 do 100) a smer riadenia (od -100 do 100)

C. Ukážkové programy

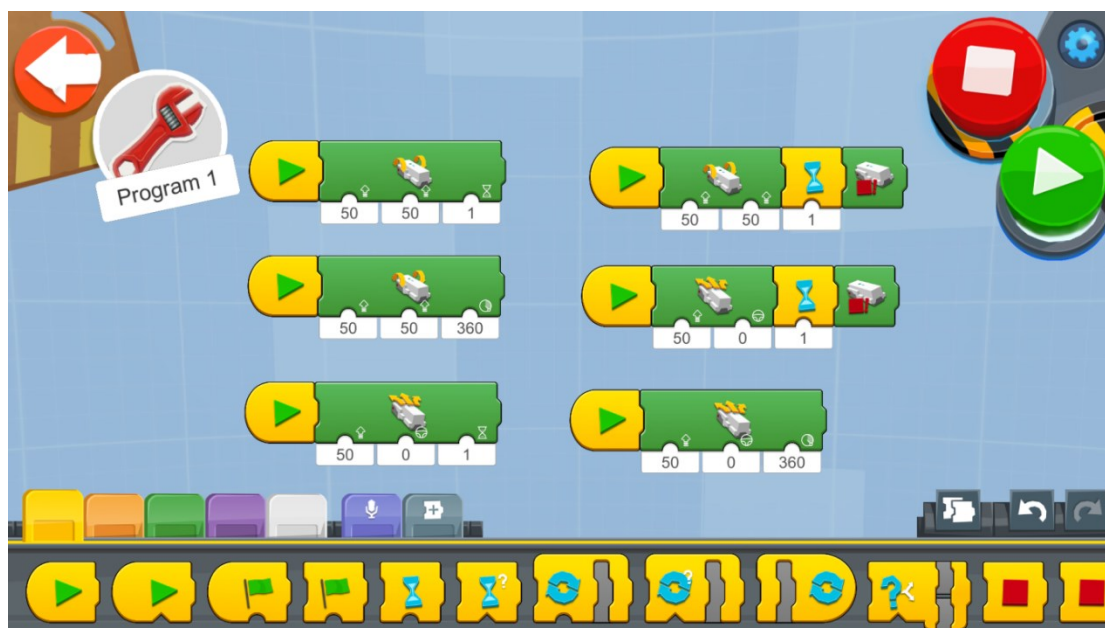


Foto 8 – Tvorba SMARC pohybov.

Skúste použiť rôzne bloky na pohyb ako v príklade vyššie uvedeného programu

Video s nasledujúcim textom a programom Smarc vykonávajúcim programy.

Vytvorte nový projekt na Creative Canva a spustite každý blok kódu a všimnite si rozdiely a podobnosti.

Upozorňujeme, že pohybujúce sa bloky, ktoré neurčujú časový interval alebo počet stupňov (rotácií) pohybu, budú motory otáčať integrovane, pokiaľ nie je pridaný blok čakania, ktorý nastaví časový interval pre prácu motorov, po ktorom nasleduje zastavenie blok motorov.



Naučte sa, ako urobiť Smarc v pohybe. - <https://youtu.be/-Jor9F0y4fA>

Poznámka: ak študent dokončil prvú aktivitu jednoduchým autom alebo prvou zo série Vernies, môže použiť zjednodušenú verziu pohybových blokov.



Foto 9- Hlavná scéna v hale aplikácie Lego Boost

Krok 1: Prejdite na ikonu Simple Car (jednoduché auto) na hlavnej scéne v hale a kliknite na ňu



Foto 10- Aktivita Simple Car (jednoduchého auta) 1

Krok 2: Kliknite na prvú aktivitu

TIP pre učiteľa: Programy môžete odomknúť stlačením aktivity, ktorú chcete odomknúť, na päť sekúnd

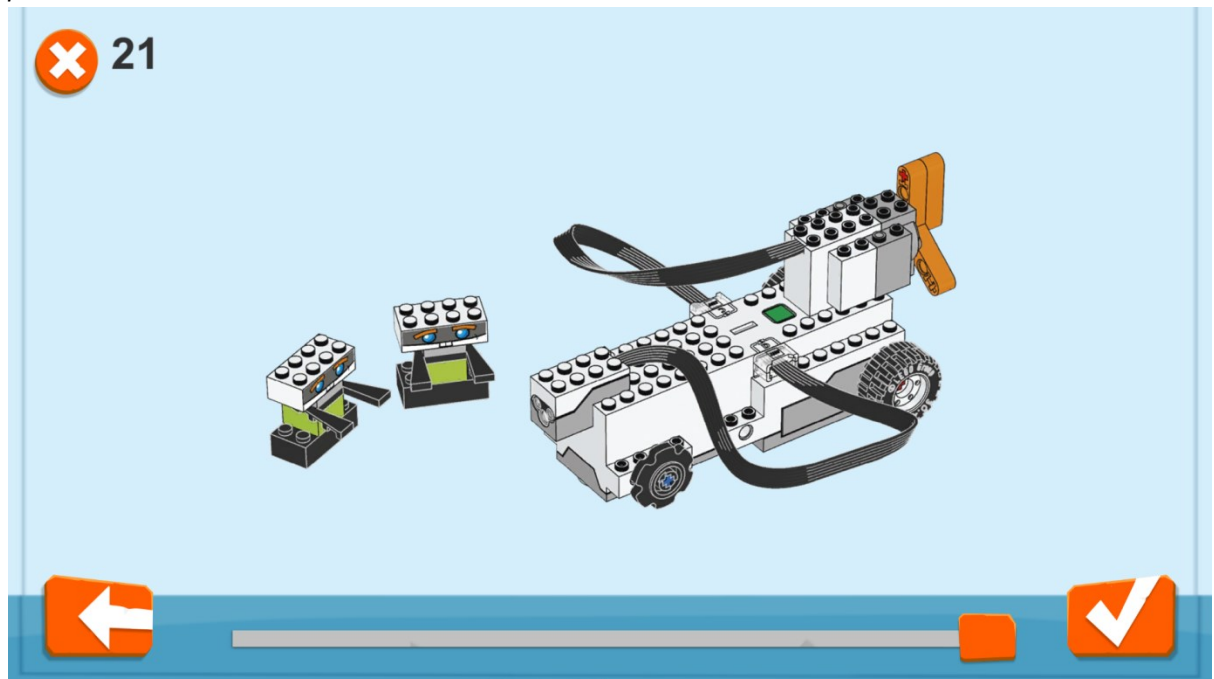


Foto 11- Stavba jednoduchého auta

Krok 3: Ak sa jedná o prvé otvorenie tejto konkrétnej činnosti, ukázal sa podrobný sprievodca pre výrobu automobilu. Vždy sa môžete len posunúť na poslednú stránku a kliknúť na ikonu zaškrtnutia.

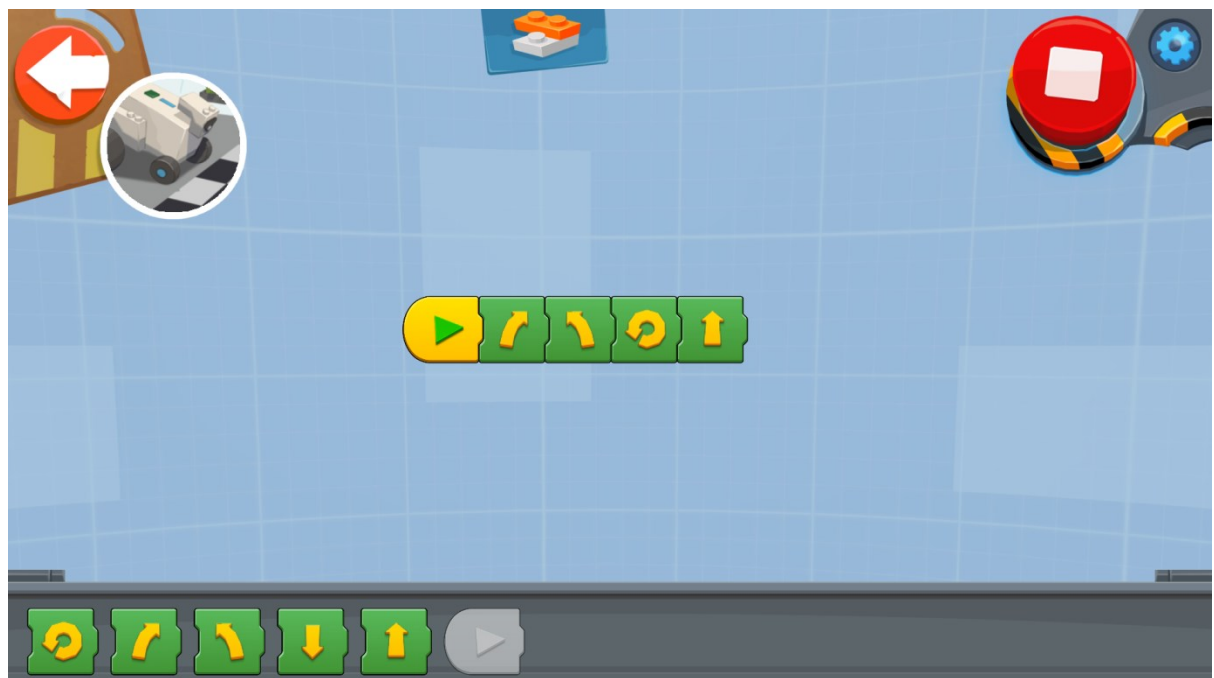


Foto 12- Základné bloky pre jednoduché pohyby

Krok 4: Pomocou jednoduchých pohybových blokov pohybujte Smarcom. Ako ďaleko sa posunie, keď sa použije jeden blok so šípkami? Pohybuje sa očakávaným smerom?

Poznámka pre učiteľa: Hnacie motory pre Smarc sú v prednej časti robota, na vozidle Simple Drive Car (auto s jednoduchým riadením) v zadnej časti robota.

III. Koncept kódovania slučky

Smyčky umožňujú spoločnosti Smarc ľahko opakovať bloky kódu bez toho, aby bolo treba opakovane vkladať bloky kódu. V kódovaní sú rôzne typy slučiek.

Program „For Loops“ (pre slučky) umožňuje programátorom opakovať kód určitý počet opakovaní.

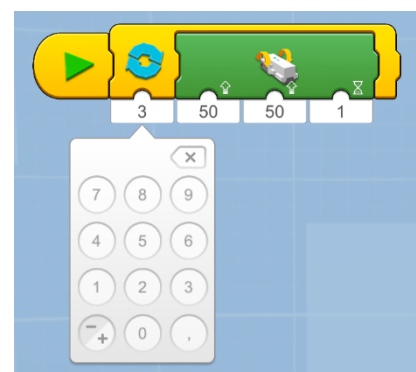
„While Loops“ (zatiaľ, čo slučky) vyžaduje, aby bola podmienka pravdivá, aby sa mohol opakovať kód.

Program „Forever Loops“ (navždy slučky) opakuje kód navždy a malo by sa mu vyhnúť, ak to nie je potrebné, pretože môže spotrebovať všetky zdroje nášho počítača a vybiť batérie nášho robota.

- **“For Loops” (pre slučky).**

The Loop For Count (slučka pre počet): Slučky uzavretých blokov kódu pre konkrétny počet opakovaní, ktoré môže nastaviť užívateľ

Video



- **“While Loops” (zatiaľ čo slučky...)**

Loop While True (slučka, zatiaľ čo je pravda): Slučky uzavretých blokov kódu, zatiaľ čo podmienka je pravdivá.

Užívateľ môže nastaviť, či sa slučka vykoná, ak je podmienka pravdivá alebo nepravdivá.

Video



- **“Forever Loops” (navždy slučky)**

Loop Forever (slučka navždy): Slučky priložených blokov kódu





konceptia kódovania slučky - <https://youtu.be/hPpylPF2QZM>

Ukážkové programy blokov slučiek

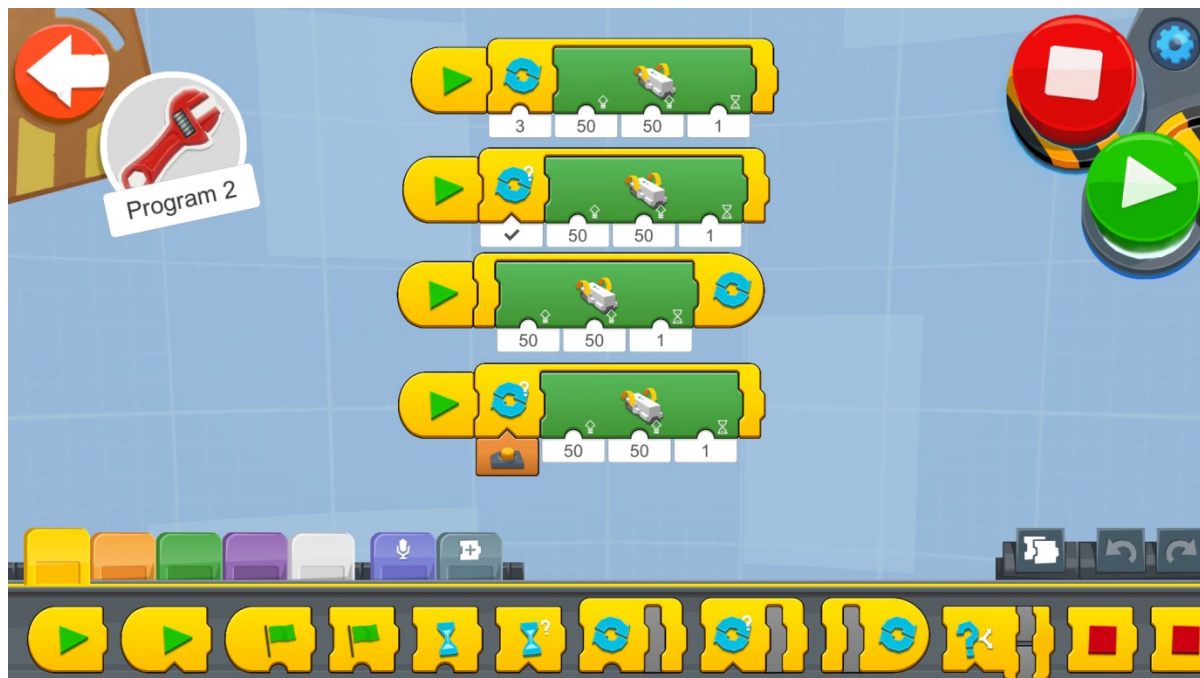


Foto 13-Program – Blok slučiek

Skúste použiť rôzne bloky pre príkazy slučiek, ako je tomu v príklade vyššie.

Vytvorte nový projekt na Creative Canva a spustite každý blok kódu a všimnite si rozdiely a podobnosti.

Všimnite si, že posledný blok kódu nefunguje správne. Vyskúšajte to tak, aby po stlačení tlačidla došlo k aktivácii slučky (podmienka slučky sa stane pravdivou).



slučkové ukážkové programy - <https://youtu.be/PcGk9zD2M5k>

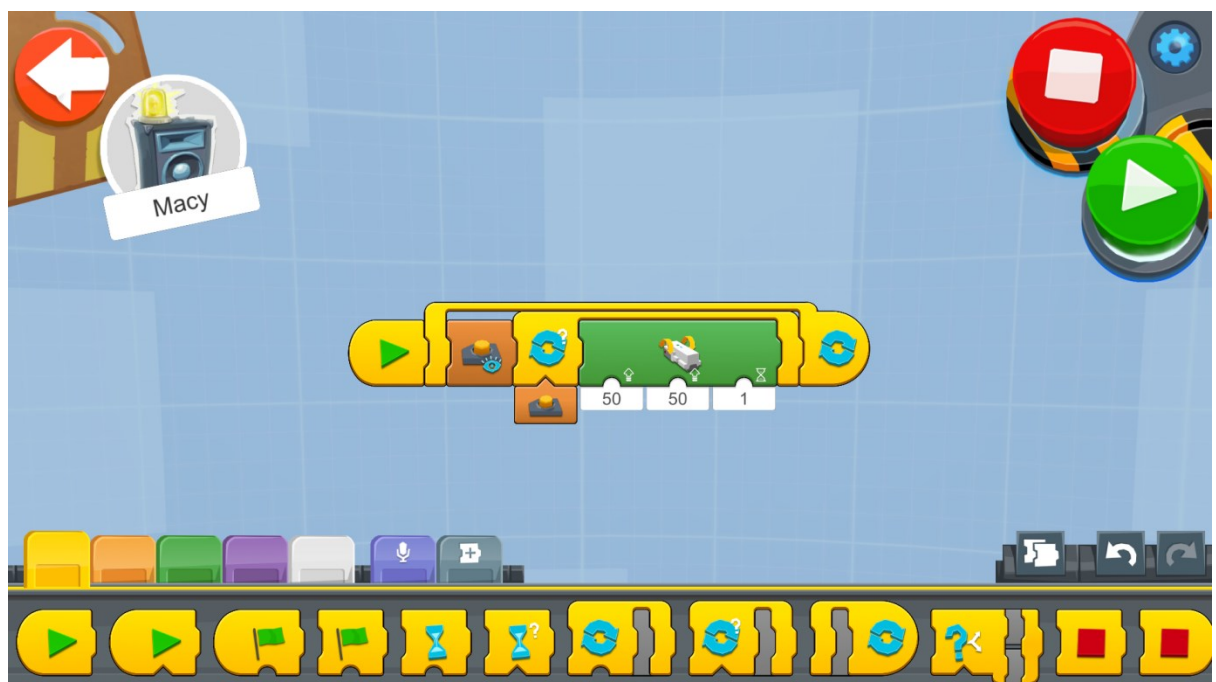


Foto 14-Program – vnorená slučka

Riešením je vložiť ďalšiu slučku Loop Forever (*slučka navždy*) a pridať blok vzhľadu tlačidiel mimo blok Loop While True (*slučka, zatiaľ čo je pravda*). Toto sa nazýva nested loop (*vnorená slučka*) - (slučka vo vnútri slučky).

2. Používanie senzorov so Smarcom

I. Sensory

V tejto časti uvidíme, ako môžeme zabudovať senzor do Smarc a ako môžeme tento senzor využiť na detekciu prekážok, rôznych farieb, intenzity svetla, ako aj vzdialenosti od objektu.

A. Lego Boost senzor farieb a vzdialenosti

Senzor je zariadenie, ktoré detekuje udalosti alebo zmeny v okolitom prostredí a odosiela informácie do výpočtového zariadenia, aby ich mohol spracovať a vykonať akcie. V našom každodennom živote existuje veľa príkladov použitia senzorov. Od pohybového senzora, ktorý deteguje pohyb a rozsvieti svetlá alebo otvorí posuvné dvere, až po teplotné senzory, ktoré detegujú teplotu a zapnú ohrievače a podmienky vzduchu. Sensory sa používajú aj vo väčšine typov diaľkovo ovládaných zariadení, napríklad v televízoroch a klimatizáciách.

LEGO BOOST senzor vzdialenosti a farieb, ako je znázornené na obrázku, sa používa na:

- Sensing distance (*snímaná vzdialenosť*) - Ako ďaleko sa nachádza prekážka od snímača.
- Sensing color (*snímajúca farba*) - dokáže zistiť konkrétne farby (Black-čierna, Blue-modrá, Green-zelená, Yellow-žltá, Red-červená, White-biela), ako aj absenciu farby.
- Detecting movement (*detekcia pohybu*) - môže byť použitá ako detektor pohybu
- Detecting light (*detekcia svetla*) - dokáže detekovať úroveň svetla odrazeného od senzora od 0 - najtmavšieho po 10 - najsvetlejšieho.

Senzor farieb a vzdialenosti môže navyše vyžarovať rôzne farby (Red-červená, Green-zelená, Blue-modrá a ich kombinácia troch).



Senzor farieb a vzdialenosti

B. Detekcia objektov

The Detecting Objects Blocks (*blokovanie detekujúcich objektov*)



Trigger on Distance Block (*spúšť na bloku vzdialenosti*) - Spustí sa, keď je vzdialenosť určená snímačom menšia ako vzdialenosť označená hodnotou pod ním. Keď je spustený, vykoná nasledujúcu postupnosť kódu. Hodnoty môžu byť od 0 do 10..



Sensor Distance Reporter (*reportér vzdialenosti snímača*) - Zobrazuje aktuálnu vzdialenosť určenú snímačom v reálnom čase. Musí byť pripojený k spodnej časti ostatných blokov, aby mohol byť zahrnutý do softvéru. Hodnoty môžu byť od 0 do 10.



Wait for Distance (*čakanie na vzdialenosť*) - Čaká na to, aby bola vzdialenosť určená snímačom menšia ako vzdialenosť uvedená hodnotou pod ním. Ak objekt nie je bližšie ako uvedená hodnota, program zostane pozastavený a program začne dodržiavať nasledujúcu postupnosť pokynov, kým nebude splnená podmienka. Hodnoty môžu byť od 0 do 10.

Ukážkové programy zisťovania prekážok

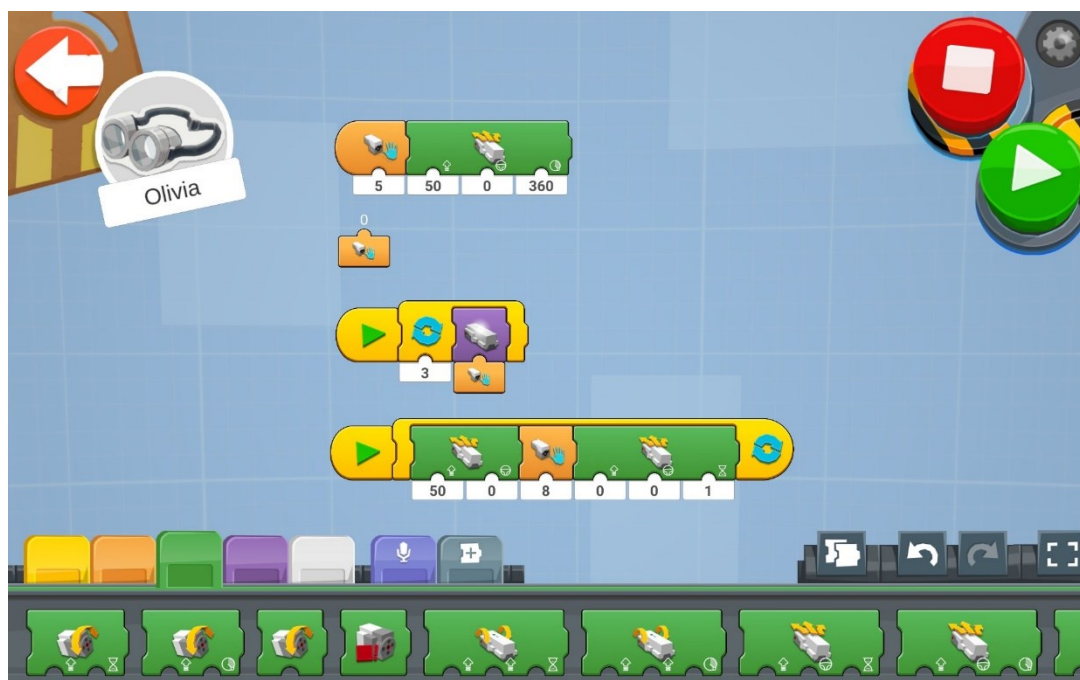


Foto 15 - Zisťovanie prekážok

1. Prvý program sa aktivuje, keď je objekt menej ako vzdialenosť 5 pred Smarcom. Smarc sa potom bude pohybovať o 360 stupňov dopredu rýchlosťou 50.

2. Druhý blok iba indikuje v reálnom čase vzdialenosť od objektu pred snímačom.

3. Tretí program berie vzdialenosť a používa ju ako vstup na výber a zobrazenie farieb na žiarovke movehub. Všimnite si, že keď sa zmení vzdialenosť od objektu, zmení sa aj farba.

Vo štvrtom programe sa Smarc začne pohybovať rýchlosťou 5 dopredu a keď zistí objekt vo vzdialenosti menšej ako 8, na chvíľu sa zastaví, potom sa pokúsi pohnúť znova..



Zisťovanie prekážok - <https://youtu.be/CAXnOmMiUDQ>

C. Zisťovanie farieb

Senzor LEGO BOOST dokáže detekovať šesť farieb (Black-čierna, Blue-modrá, Green-zelená, Yellow-žltá, Red-červená, White-biela a absencia farby - žiadna farba, čo znamená, že senzor nezistí žiadny predmet).

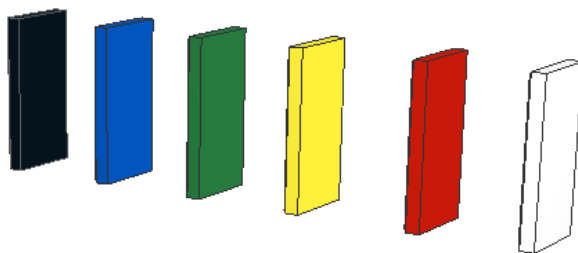


Foto 16- Farby, ktoré senzor vidí: čierna, modrá, zelená, žltá, červená, biela

Poznámky:

Ak sa objekt nenachádza v navrhovanom rozsahu snímača, farba nemusí byť detekovaná správne. Napríklad zelený objekt môže byť detekovaný ako modrý.

Pokiaľ farba objektu nie je jednou zo šiestich detekovateľných farieb, aplikácia ju zobrazí ako najbližšiu možnú farbu. Napríklad oranžová bude prezentovaná ako červená.

The Detecting Colors Blocks (*detekčné bloky farieb*) :



Trigger on Color (*spúšť na farbe*) - Spustí sa, keď sa farba nameraná snímačom rovná farbe znázornenej hodnotou pod ním. Po aktivácii sa vykoná nasledujúca sekvencia kódu. Môže sa použiť sedem hodnôt: žiadna farba, Black-čierna, Blue-modrá, Green-zelená, Yellow-žltá, Red-červená a White-biela.



Wait for Color (*čakanie na farbu*) - Čaká na to, aby sa farba určená snímačom rovnala farbe znázornenej hodnotou pod ňou. Ak zistená farba nie je rovná zadanej hodnote, program zostane pozastavený a začne sa nasledujúca postupnosť pokynov, kým nebude splnená podmienka. Môže sa použiť sedem hodnôt: žiadna farba, čierna, modrá, zelená, žltá, červená a biela.



Sensor Color Reporter (*reportér snímača farieb*) - Zobrazuje aktuálnu farbu nameranú snímačom v reálnom čase. Aby sa dal použiť v programe, musí byť umiestnený na spodnej časti ostatných tehál. Uvidíte sedem hodnôt: žiadna farba, čierna, modrá, zelená, fialová, červená a biela.

Ukážkové programy na zisťovanie farieb

Predtým, ako uvidíme programy na detekciu farieb, najskôr si musíme nainštalovať interaktívny motor.

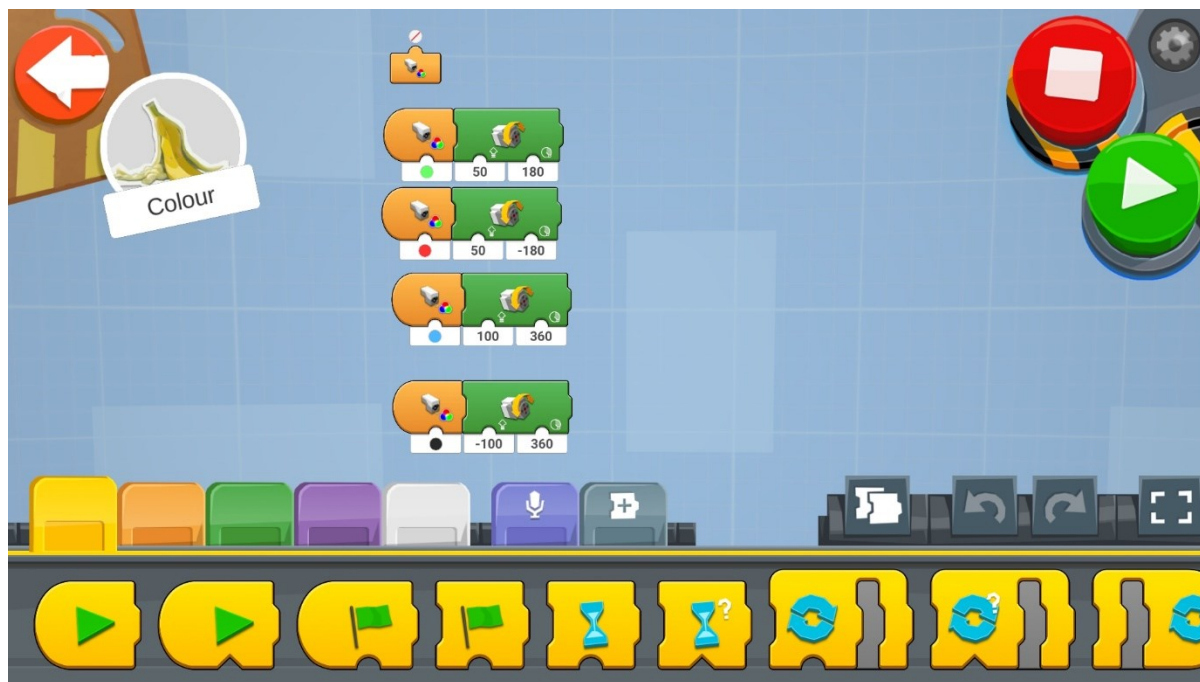


Foto 17- Program -detekcia farieb

1. Prvý blok iba indikuje v reálnom čase farbu objektu pred snímačom.
2. Druhý program sa aktivuje, keď je objekt zelenej farby pred senzorom spoločnosti Smarc. Spoločnosť Smarc potom pohne vrtuľou pri rýchlosti 50 o 180 stupňov doprava.
3. Tretí program sa aktivuje, keď je objekt červenej farby pred senzorom spoločnosti Smarc. Potom spoločnosť Smarc posunie zadnú vrtuľu o 180 stupňov doľava pri rýchlosti 50.
4. Štvrtý program sa aktivuje, keď je objekt modrej farby pred senzorom Smarc. Spoločnosť Smarc potom pri rýchlosti 100 posunie zadný vrtule o 360 stupňov doprava.
5. Piaty program sa aktivuje, keď je objekt zelenej čiernej farby pred senzorom spoločnosti Smarc. Aplikácia Smarc potom posunie zadnú vrtuľu o 360 stupňov doľava pri rýchlosti 100.



Trigger on Color
Triggers when the color measured by the sensor is equal to the color shown by the value below it. When activated, the following code sequence is executed. Seven values can be used: No color, Black, Blue, Green, Yellow, Red and White.



Detekcia farieb - <https://youtu.be/onDfjAmNHio>

D. If/Else bloky:

Príkaz If / Else (Ak / Inak) umožňuje spoločnosti Smarc ľahko sa rozhodovať na základe vstupov zo senzora. Ak je podmienka splnená (to znamená, že je PRAVDA), program Smarc vykoná konkrétny blok kódu. Ak nie je podmienka splnená (čo znamená, že je NEPRAVDA), program Smarc vykoná ďalší blok kódu..



If/Else (Ak/inak)- ak je podmienka True, vykonajte hornú sekvenciu, inak ak je nepravda, vykonajte spodnú sekvenciu.



Equal Operator (*rovnaký operátor*) - vráti hodnotu True, keď sa vstup zo snímača (farba / vzdialenosť / okolité svetlo) rovná hodnote.



Less Than Operator (*menej ako operátor*) - vráti hodnotu True, ak je vstup zo snímača (farba / vzdialenosť / okolité svetlo) menší ako hodnota.



Greater Than Operator (*viac ako operátor*) - vráti hodnotu True, ak je vstup zo snímača (farba / vzdialenosť / okolité svetlo) väčší ako hodnota.



Not Equal Operator (*nie je rovnocenný operátor*) - vráti hodnotu True, ak sa vstup zo snímača (farba / vzdialenosť / okolité svetlo) nerovná hodnote

Meranie množstva odrazu svetla: Bloky odrazu meracieho svetla:



Sensor Light Level Reporter (*reportér úrovne snímača osvetlenia*) – V reálnom čase zobrazuje aktuálnu hladinu okolitého svetla nameranú snímačom. Aby bolo možné ho použiť v programe, musí byť pripojený k spodnej časti ostatných blokov. Môže označovať hodnoty od jednej (1) do desať (10). Jeden je najtmavší a desať najjasnejších.

II. Detekcia zvuku pomocou protokolu Smarc

A. Reakcia na zvuk

Zvukom aktivovaný robot

V tejto časti uvidíme, ako dokážeme ovládať Smarc zvukmi, ako je napríklad tleskanie rukami, a použitím rôznych intenzít zvuku pomocou blokov zvukového senzora.

V prípade Smarc je zvukovým senzorom vlastne zabudovaný mikrofón, ktorý sa nenachádza v Move Hub. Namiesto toho sa spolieha na mikrofón, ktorý je umiestnený vo vnútri zariadenia, ku ktorému je Smarc pripojený, napríklad k tabletu, notebooku alebo smartfónu.

Aplikácia Lego Boost rozpozná, aký hlasný zvuk je. Preto ho môžeme naprogramovať tak, aby reagoval rôzne v závislosti od prijatej intenzity zvuku.

B. Bloky zvukového senzora:



Trigger on Sound Level (*spúšťač na úrovni zvuku*) - Spustí sa, keď je hladina zvuku nameraná snímačom vyššia ako hladina zvuku znázornená hodnotou pod ním. Po spustení vykoná nasledujúcu sekvenciu kódov. Bude mať 11 hodnôt od 0 do 10.



Wait for Sound Level (*čakanie na úroveň zvuku*) - Čaká, kým intenzita zvuku nameraná snímačom nebude vyššia ako hladina zvuku uvedená nižšie. Ak zaregistrovaná hladina zvuku nie je vyššia ako zadaná hodnota, program zostane pozastavený a začne sa nasledujúca séria pokynov, kým nebude splnené dané kritérium. Vezmete jedenásť hodnôt od 0 do 10.



Sound Level Reporter (*reportér úrovne zvuku*) - Zobrazuje najnovšiu zvukovú frekvenciu vypočítanú snímačom v reálnom čase. Aby sa dal použiť v programe, musí byť pripojený k spodnej časti ostatných tehál. Bude predstavovať hodnoty od 0 do 10 vrátane jedného desatinného čísla, napr. 7,8.

Ukážkové programy na detekciu zvuku

Program 1

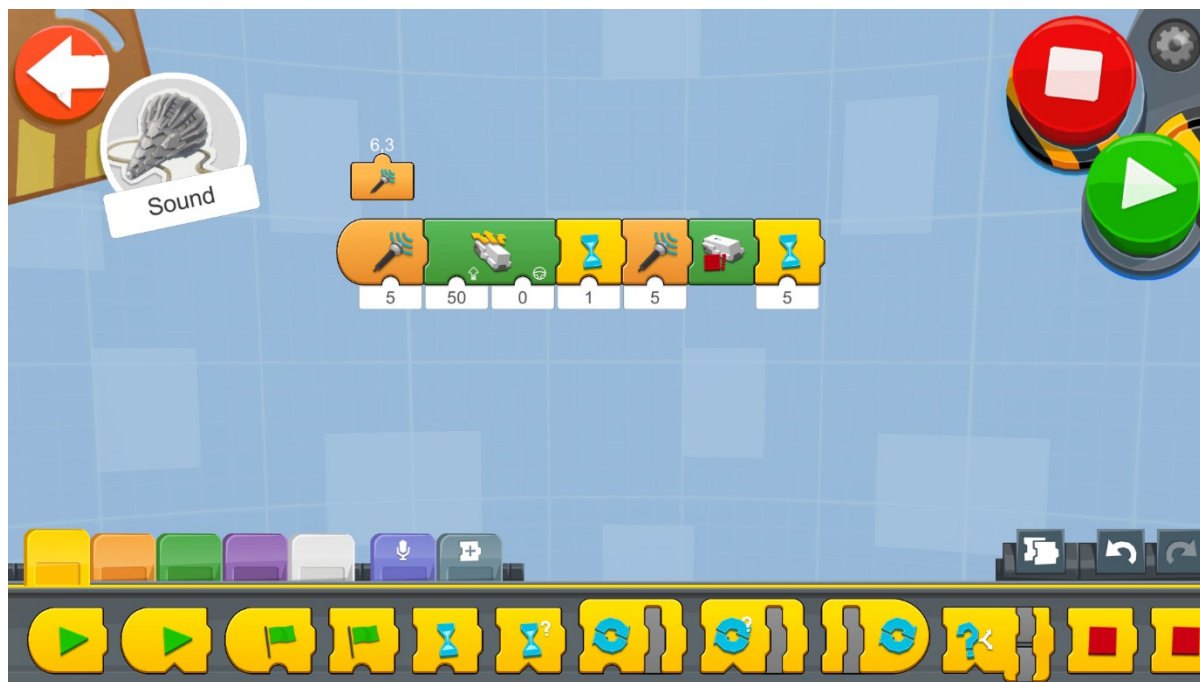


Foto 18-Program 1- Detekcia zvuku

Vytvorte nový projekt na kreatívnom plátne a spustíte blok kódu:

1. Kód sa spustí, keď sa spustí blok Trigger on Sound Level (*spúšťač na úrovni zvuku*). Stane sa to, keď sa zistí šum na úrovni vyššej ako 5.
2. Ak je detekovaný zvuk, Smarc sa pohybuje dopredu pomocou bloku Drivebase Move Steering (*Drivebase posunu riadenia*) rýchlosťou 50.
3. Blok Wait for Time (*čakanie na čas*) je nastavený na 1 sekundu. Dôvodom tohto bloku je, aby sa Smarc nezamieňal s rovnakým zvukom, aký sa používa na aktiváciu programu na posun vpred a na spustenie zvyšku kódu.
4. Blok Wait for Sound Level (*čakanie na úroveň zvuku*) čaká na šum na úrovni vyššej ako 5, aby spustil vykonanie zvyšku kódu.
5. Nakoniec, keď sa aktivuje blok Wait for Sound (*čakanie na zvuk*), aktivuje sa Drivebase Stop a zastaví Smarc v pohybe.



Detekcia zvuku 1 - <https://youtu.be/3xrI2SGrlh8>

Program 2



Foto 19- Program 2 – Detekcia zvuku

Vytvorte nový projekt na kreatívnom plátne a spustíte blok kódu:

1. Infinite Loop (*nekonečná slučka*) bude spúšťať kód opakovane, aby Smarc neustále kontroloval čítanie zo senzora a podľa toho konal.
2. Vo vnútri slučky sa nachádza blok If / Else (*ak/inak*), ktorý, ak má hodnotu TRUE, sa Smarc pohne dopredu a ak je FALSE, Smarc sa zastaví.
3. Blok Compare Greater Than (*porovnať väčšie ako*) nastavuje podmienky pre blok Prepínač. Blok Compare Greater Than (*porovnať väčšie ako*) porovnáva výstup bloku Sensor Sound Level Reporter (*reportéra snímača úrovne zvuku*) s hodnotou 2. Vráti hodnotu true, keď snímač zistí hladinu zvuku vyššiu ako 2, a hodnotu false, ak je rovný alebo menší ako 2.
4. Ak je výstup bloku If / Else (*ak/inak*) pravdivý, blok Drivebase Move Steering (*Drivebase riadenia pohonu*) presunie Smarc rýchlosťou 55, v uhle 0 stupňov, a bude to robiť ešte 0,2 sekundy. Ak je nepravdivé, blok Drivebase Stop zastaví Smarc v pohybe.



https://youtu.be/oMq_olyajE0

III. Navigácia v Smarc pomocou diaľkového ovládača

A. Diaľkové ovládanie

V tejto časti uvidíme, ako sa dokáže Smarc pohybovať pomocou diaľkového ovládača.

Existuje mnoho rôznych spôsobov komunikácie s robotom. Roboty sa najčastejšie ovládajú pomocou káblového alebo bezdrôtového ovládača alebo môžu bežať autonómne získavaním pokynov od svojho riadiaceho programu (čo sme robili so programom Smarc v predchádzajúcich kapitolách).

Diaľkovo ovládané roboty môžu mať rôzne vedecké použitia vrátane nebezpečných prostredí, ako sú extrémne horúčavy alebo chladu a rádioaktívne miesta, práce v hlbokom oceáne na umiestňovanie potrubí a káblov, ako aj prieskum vesmíru a planét.

B. Bloky diaľkového ovládania:



Joystick Widget Show (zobrazenie miniaplikácie Joystick) - Keď je tento blok aktivovaný, predstavuje miniaplikácia Joystick v aplikácii Lego BOOST.



Joystick Widget Hide (*skrytá miniaplikácia Joystick*) - Keď je tento blok aktivovaný, skryje miniaplikácia Joystick z aplikácie Lego BOOST.



Joystick Widget Speed Reporter (*rýchlosť joysticku – reporter*) - Zobrazí v reálnom čase aktuálnu rýchlosť Joystick Widget (-100..100). Aby bolo možné ho použiť v programe, musí byť pripojený k spodnej časti ostatných blokov. Vo väčšine prípadov sa používa ako vstup riadenia do bloku riadenia Drivebase Move. Môže zobrazovať hodnoty od -100 do 100 vrátane dvoch desatinných hodnôt, napríklad 78,89.



Joystick Widget Steering Reporter (*reportér riadenia joysticku*) - Zobrazuje aktuálne riadenie Joystick Widget v reálnom čase (-100..100). Aby bolo možné ho použiť v programe, musí byť pripojený k spodnej časti ostatných blokov. Vo väčšine prípadov sa používa ako vstup rýchlosti do bloku riadenia Drivebase Move. Môže zobrazovať hodnoty od -100 do 100 vrátane dvoch desatinných hodnôt, napríklad 78,89.

Ukážkové programy pre diaľkové ovládanie

Program 1



Foto 20-Program – Diaľkové ovládanie

Vytvorte nový projekt na kreatívnom plátne a spustíte blok kódu:

1. Na Joystick Widget Show (miniaplikácii Joystick) sa zobrazí ovládač joysticku v aplikácii (pozri obrázok).
2. Infinite Loop (nekonečná slučka) spustí kód opakovane, takže Smarc po celú dobu kontroluje čítanie z joysticku a koná podľa toho.
3. Vo vnútri slučky sa nachádza blok Drivebase Move Steering (Drivebase riadenia pohonu), ktorý bude brať ako rýchlosť načítanie čítania bloku Joystick Widget Speed Reporter (miniaplikácie rýchlosti joysticku – reportér) a ako smerovanie načítanie bloku Joystick Widget Steering Reporter (miniaplikácie riadenia Joysticku)



Photo 21-Program - Joystick Controller

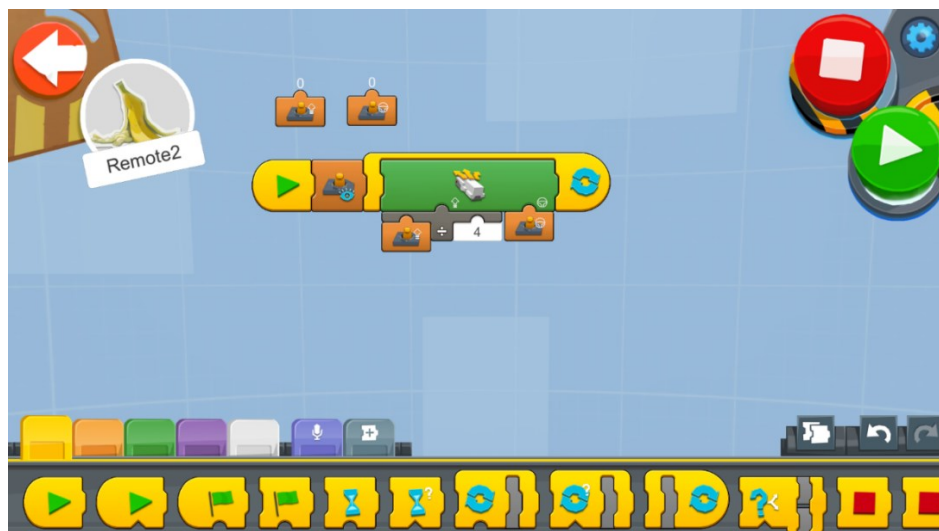
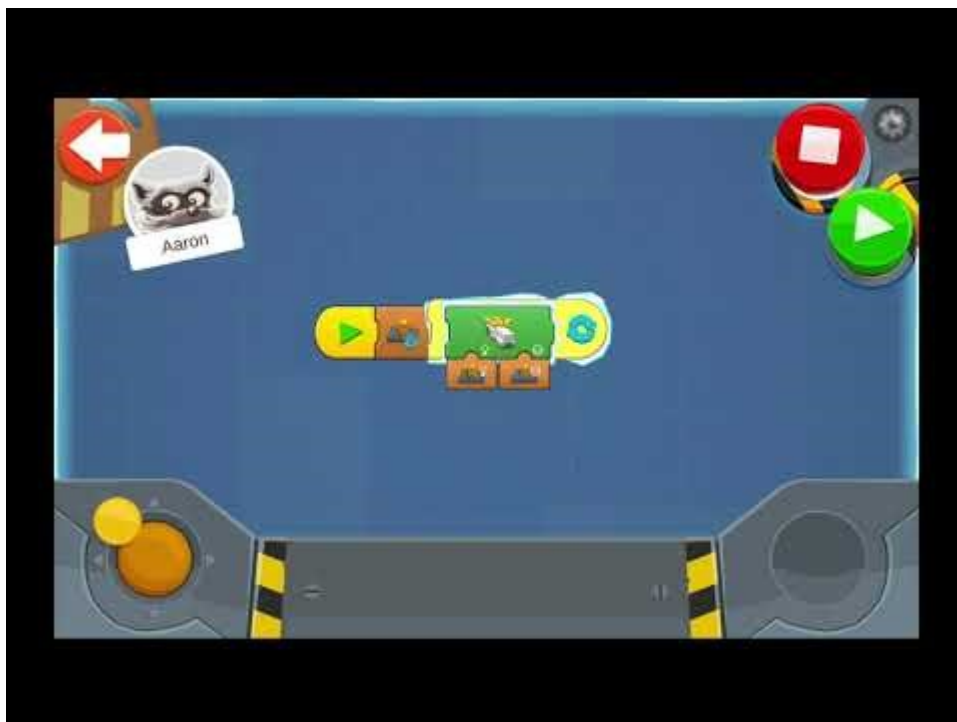


Foto 22- Program 2 – Diaľkové ovládanie

Niekedy potrebujeme presnejšiu kontrolu nad Smarcom a vstupná rýchlosť je na to príliš vysoká. Na tento účel môžeme urobiť trik, aby sme znížili vstupnú rýchlosť, ktorú odosiela blok Joystick Widget Speed Reporter (miniaplikácie reportéra rýchlosti joysticku). Na zadanie rýchlosti bloku Drivebase Move (pohon) sa používa operátor divízie. Vstup bloku Joystick Widget Speed Reporter (miniaplikácie reportéra rýchlosti joysticku) sa najskôr vydolí 4 a potom sa hodnota odošle ako hodnota rýchlosti. Napríklad, ak je hodnota načítaná z joysticku 100, rýchlosť bude:

$$100 \div 4 = 25$$

Skutočná rýchlosť by teda bola 25..



Diaľkové ovládanie - <https://youtu.be/Tswdypjm7bk>



Tento dokument sa môže kopírovať, rozmnožovať alebo upravovať v súlade s vyššie uvedenými pravidlami. Okrem toho musí byť jasne uvedené potvrdenie o autoroch dokumentu a všetky príslušné časti oznámenia o autorských právach.

Všetky práva sú vyhrazené.

Copyright © in ROBOTICS4DEAF consortium, 2019-2021

