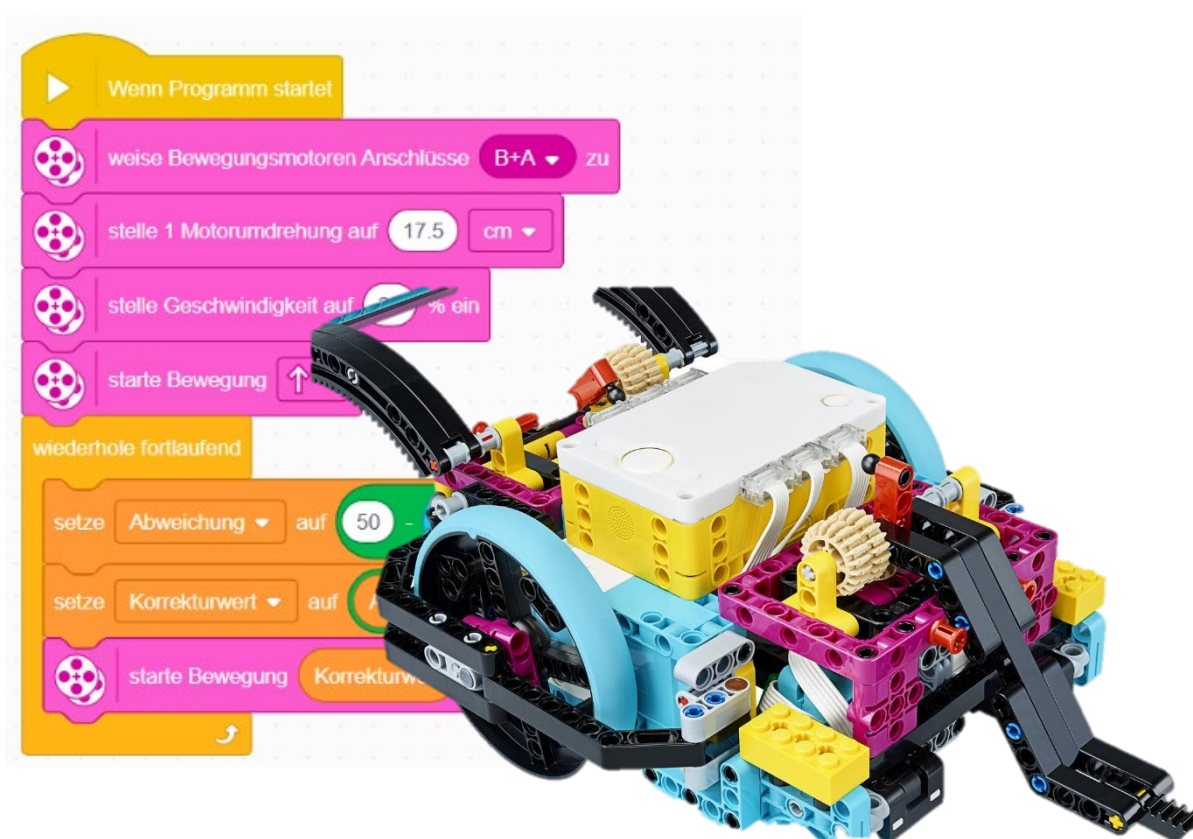




SWITZERLAND

Robotica per principianti

Dossier di apprendimento



di Vera Hausherr
Versione 2.0 /11.09.2023

Copyright

Questo dossier di apprendimento è proprietà intellettuale di Vera Hausherr e della World Robot Olympiad Switzerland (WRO). Viene fornito gratuitamente ai partecipanti del workshop WRO e può essere utilizzato da loro, dai colleghi e dalle colleghe della squadra o dagli studenti e dalle studentesse. Invitiamo tutti i partecipanti a rispettare la proprietà intellettuale e a non inoltrare il dossier di apprendimento ad altre persone.

Questa opera non può essere inoltrata a titolo oneroso.

Marchi

LEGO®, SPIKE™ Prime, LEGO® Education, SPIKE™ App e altri termini collegati a LEGO sono marchi registrati. Per rendere il testo più leggibile i simboli del marchio registrato vengono omessi.

Nota tecnica

Questo dossier di apprendimento è stato redatto sulla base della versione 3 dell'app SPIKE. Possono esserci delle piccole divergenze fra le illustrazioni presenti nel dossier e versioni precedenti (o versioni successive non ancora rilasciate) dell'app SPIKE.

Parte 1: Cos'è la robotica?

1: Macchine, computer e robot

Obiettivo di apprendimento:

- Conosco la differenza fra macchine, computer e robot.

Inserisci correttamente gli oggetti nella tabella. Ti vengono in mente altri esempi?



Macchina	Computer	Robot



Nota

Una **macchina** può lavorare ma non pensare.

Un **computer** può pensare ma non può lavorare.

Un **robot** può sia lavorare che pensare.

2: Di quali componenti ha bisogno un robot?

Obiettivi di apprendimento:

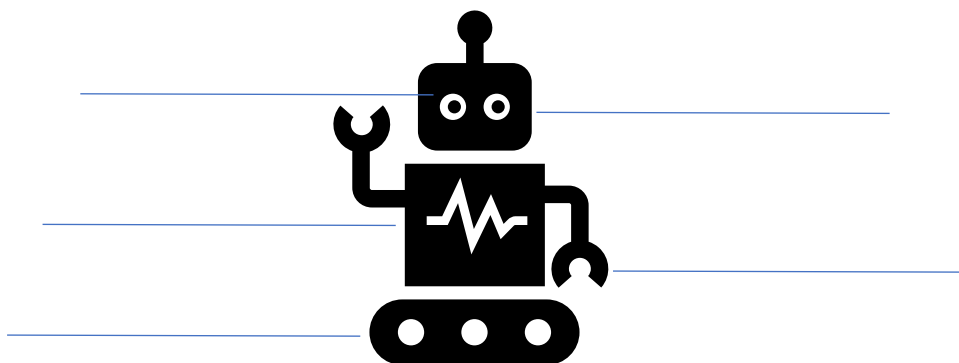
- So quali componenti fanno parte di un robot.
- So perché i sensori sono importanti.
- Conosco i componenti di SPIKE Prime.

Nell'esercizio precedente hai visto che esistono diversi tipi di robot. Ma tutti questi robot hanno alcuni componenti in comune.

2.1 Vista d'insieme

Esercizio

Associa i seguenti termini al robot: *apparato locomotore*, «cervello», *sensori*, *alimentazione elettrica*, *strumenti*.



2.2 A cosa servono questi componenti?

Esercizio

Inserisci i termini dell'esercizio 2.1 nella tabella.






Percezione dell'ambiente (ad esempio riconoscere un colore o misurare una distanza)	
Spostamento (ad esempio camminare o muoversi)	
Controllo del robot (decidere cosa deve fare dopo)	
Lavorare (ad esempio raccogliere un oggetto)	
Avere l'energia per muoversi e lavorare	

2.3 Quali componenti ci sono in SPIKE Prime?

Esercizio

Completa la tabella.

Parole da inserire: *movimenti, scura, lontano, spostarsi, colori, «cervello», chiara, sollevare, pulsanti, matrice luce, interruttore.*



	<p>Hub con batteria e sensore giroscopico</p>	<p>L'hub contiene il _____ del robot. Sulla superficie bianca si trovano la _____ ed i _____. Con il sensore giroscopico si può capire se il robot è stato girato o inclinato.</p>
	<p>Motori</p>	<p>I motori sono necessari per tutti i _____. I piccoli motori vanno bene per _____, quelli più grandi invece per _____ qualcosa.</p>
	<p>Sensori di colore</p>	<p>Con i sensori di colore si possono riconoscere i _____ e valutare quanto una cosa sia _____ o _____.</p>
	<p>Sensore di distanza</p>	<p>Con il sensore di distanza si può misurare quanto _____ si trovi un oggetto, un ostacolo o una parete.</p>
	<p>Sensore di potenza</p>	<p>Il sensore di potenza permette di riconoscere se vi è una pressione prolungata. Si può utilizzare come un _____.</p>

3: Come fa il robot a sapere cosa deve fare?

Obiettivi di apprendimento:

- Conosco i termini «programmare» e «codice».
- So dare semplici istruzioni ad un robot utilizzando il normale linguaggio.

Il «robot orso» deve arrivare dalla sua grotta al vaso di miele. Non può scavalcare i muri. E può spostarsi sulle caselle soltanto da sinistra a destra, da destra a sinistra, dall'alto in basso e dal basso in alto (non in diagonale).

								
Muro								
			Muro					
								

Esercizio

1. Disegna sulla piantina il percorso migliore.
2. Spiega al tuo compagno/compagna qual è il modo migliore in cui l'orso può arrivare al traguardo.



Nota

Affinché il robot orso possa trovare la strada, devi dargli istruzioni precise.

Ciò significa _____.

Queste istruzioni si chiamano _____.

Esercizio: completa questo codice in modo che il robot orso trovi la strada dalla grotta al vaso di miele.

(Termini possibili: «a destra», «a sinistra», «in alto», «in basso» e i numeri da 0 a 12).

- Lascia la grotta andando _____
- Vai dritto per _____ caselle.
- Ruota _____.
- Vai dritto per _____ caselle.
- Girati _____.
- Vai dritto per _____ caselle.
- Ruota _____.
- Vai dritto per _____ caselle.
- Ruota _____.
- Vai dritto per _____ caselle.

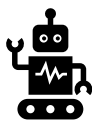
Parte 2: Introduzione alla programmazione

4: Semplificare un po' il codice: i cicli

Obiettivo di apprendimento:

- So cos'è un ciclo e a che cosa serve.

Il tuo robot deve seguire un percorso perfettamente quadrato.



Esercizio

1. Il tuo compagno/compagna interpreta un robot. Dai delle istruzioni su come deve muoversi per ottenere un percorso quadrato. Un lato del quadrato deve essere lungo esattamente come 5 piedi.
2. Cerca di scrivere un'istruzione molto breve. Deve contenere esattamente 3 righe di codice.

3. Adesso scriviamo il codice in un modo simile a quello che in seguito servirà per il nostro robot Lego: (questo non è ancora un vero codice, gli assomiglia soltanto!)



Rifletti: perché è utile usare queste ripetizioni?



Nota

Un blocco di ripetizione si chiama «**ciclo**».

A riguardo ti suggeriamo una regola d'oro per la programmazione: **non scrivere mai due volte quello che, utilizzando un ciclo, puoi scrivere una volta sola!**

4: Il robot può essere un po' più intelligente? Le condizioni.

Obiettivi di apprendimento:

- So che un robot deve prendere di continuo delle decisioni.
- So spiegare una semplice condizione.

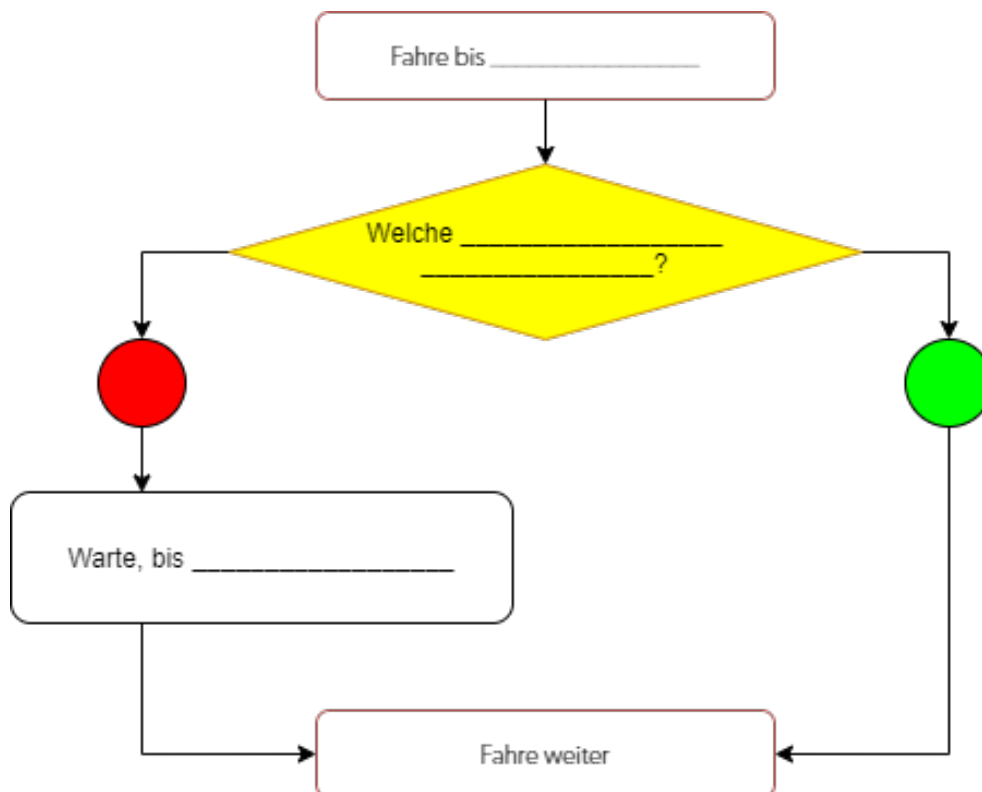
Con un semplice codice come quello del robot orso o del quadrato, un robot può già fare un po' di cose. Ma non è ancora in grado di pensare da solo. Noi però vogliamo un robot che sia davvero intelligente, che pensi da solo e decida cosa fare.



Esempio: L'auto robot nella figura qui sopra deve muoversi. Quando arriva nei pressi di un semaforo, al momento in cui si trova sulla linea gialla deve controllare di che colore è il semaforo. Se è rosso deve fermarsi fino a quando il semaforo diventa verde. Se il semaforo è verde, non deve fermarsi.

Esercizio

Completa il codice rappresentato qui come diagramma di flusso:



Queste situazioni si chiamano **decisioni** o **condizioni**. Sono la cosa più importante della programmazione!

Parte 3: Un robot Spike Prime

5: Avvio con SPIKE Prime

Obiettivi di apprendimento:

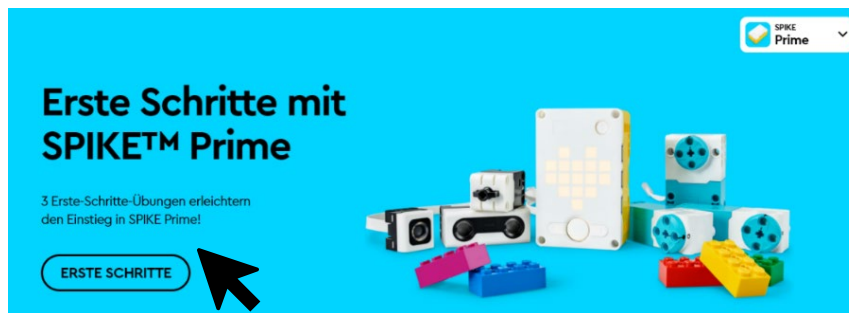
- Conosco i componenti di SPIKE Prime.
- Conosco l'app SPIKE.
- So già scrivere da solo il primo codice.

Adesso si tratta di costruire e programmare il tuo robot!

Avvia l'app **SPIKE** sul tuo computer o tablet e clicca su **Prime**.



Comincia con la sezione «**Avvio**».



Ci sono 6 esercizi con cui puoi conoscere l'hub, i motori ed i sensori e compiere quindi i primi passi nella programmazione.

Esercizio

Svolgi tutti e 6 gli esercizi. Se finisci velocemente, puoi provare anche le seguenti varianti.

Esercizio aggiuntivo 1 (matrice luce): Riesci a fare anche una freccia? Come puoi accendere o spegnere tutte le luci?

Esercizio aggiuntivo 2 (motore): Riesci a fare andare il motore anche al contrario?

Esercizio aggiuntivo 3 (sensore di colore): Prova anche con altri colori. Prendi mattoncini Lego da testare e altri oggetti di colore simile. Prova a scoprire anche quanto vicino deve essere l'oggetto al sensore per essere riconosciuto davvero bene.

Esercizio aggiuntivo 4 (sensore di distanza): Riesci ad impostare la distanza a 15 cm? Riesci a far riprodurre un altro suono?

(Non c'è un **esercizio aggiuntivo 5** poiché nel tutorial non sono disponibili abbastanza comandi!)

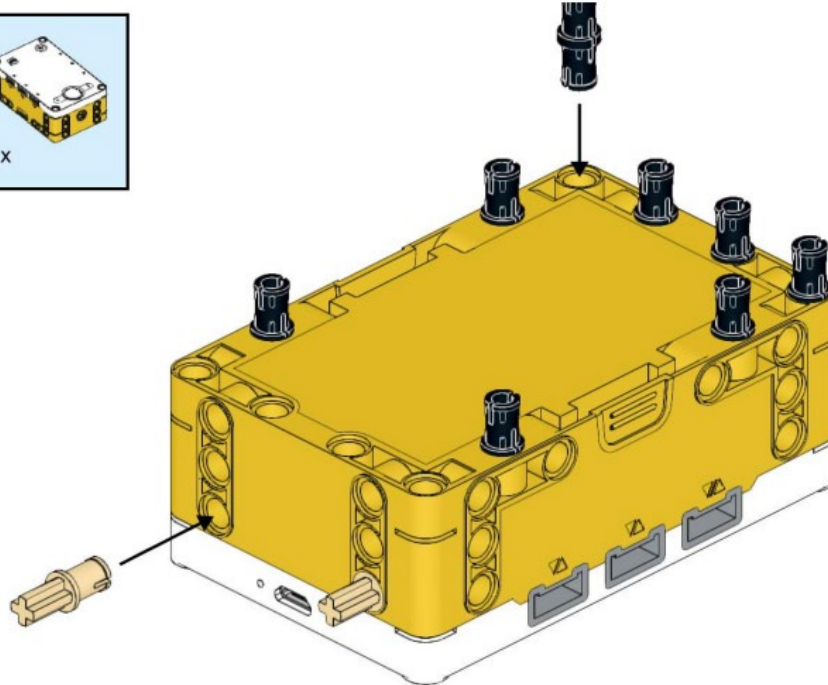
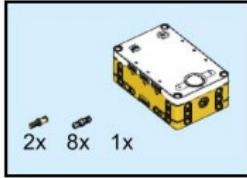
Esercizio aggiuntivo 6 (sensore giroscopico): Riesci anche a stabilire se l'hub è piegato verso avanti? E riesci a far riprodurre anche altri rumori? Provac!

6: Costruisci un piccolo robot.

Obiettivo di apprendimento:

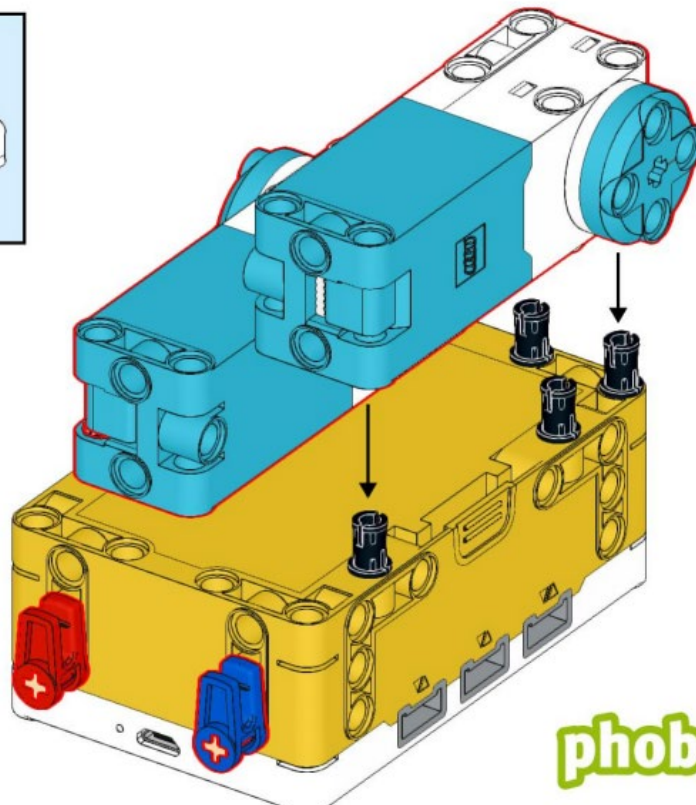
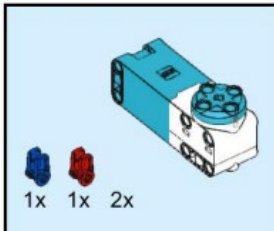
- Costruisco un piccolo robot con sensore di colore.

1



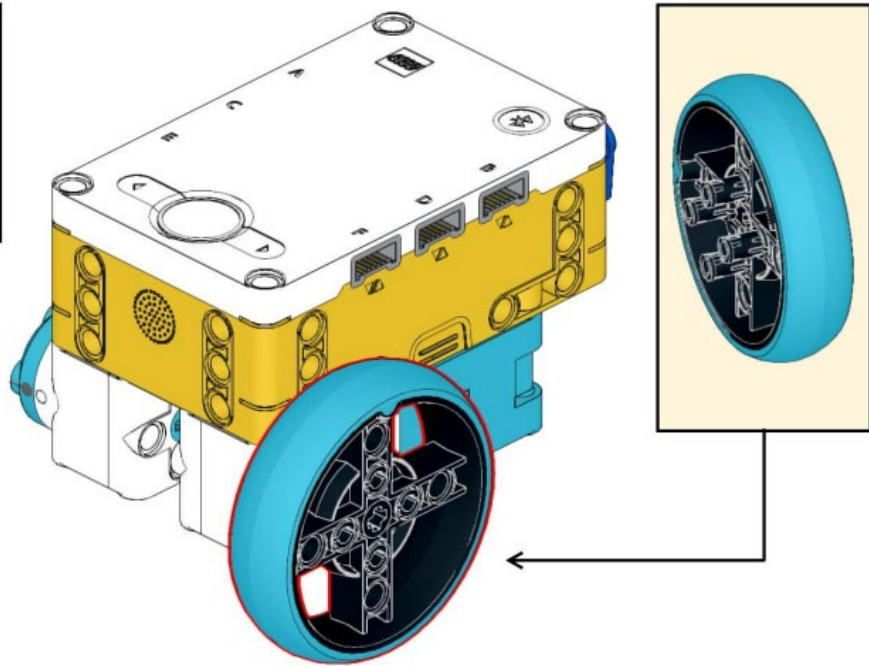
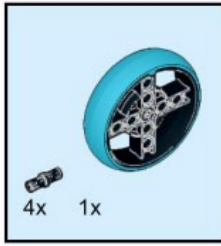
phobricks

2



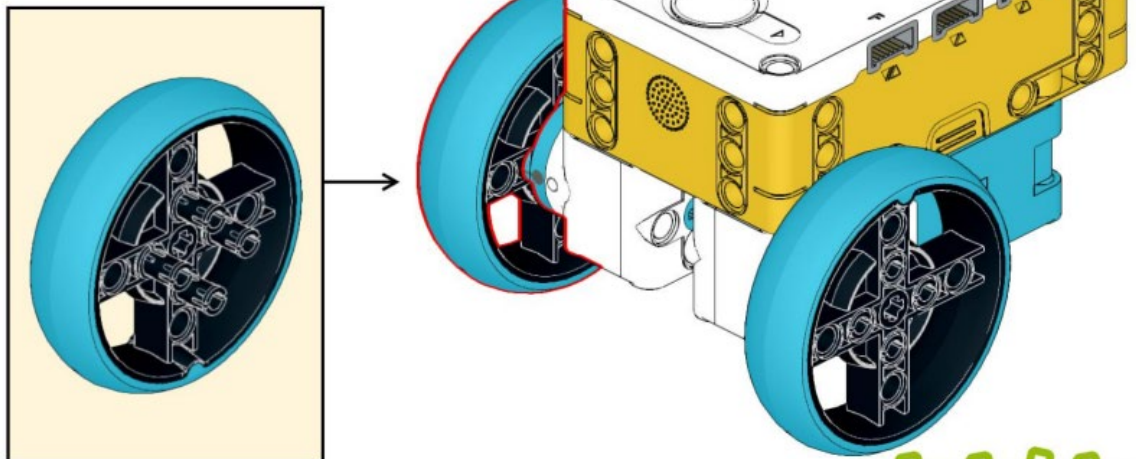
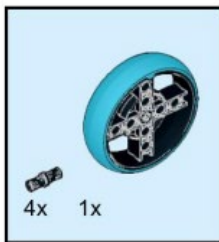
phobricks

3



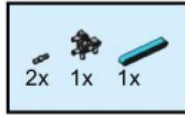
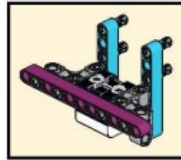
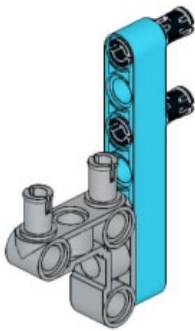
phobricks

4

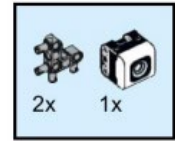
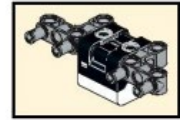
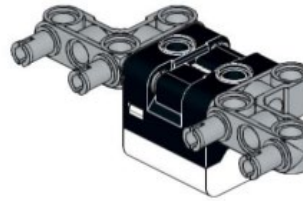


phobricks

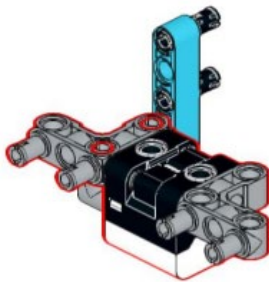
5



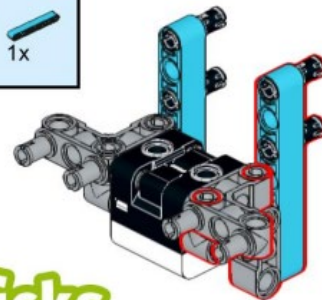
6



7

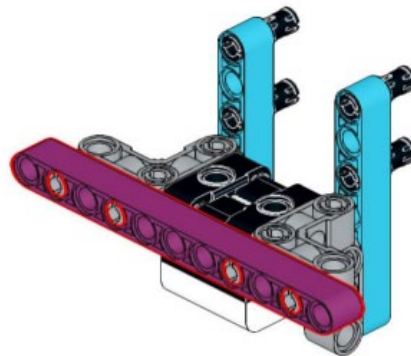
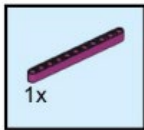


8



phobricks

9



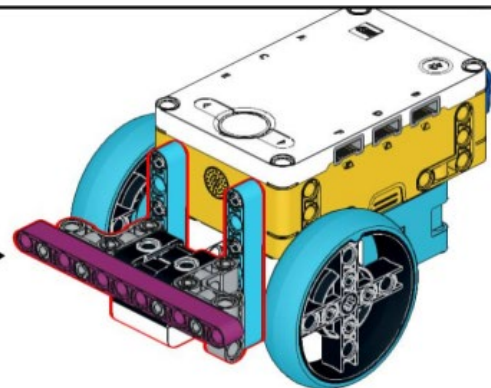
phobricks

10

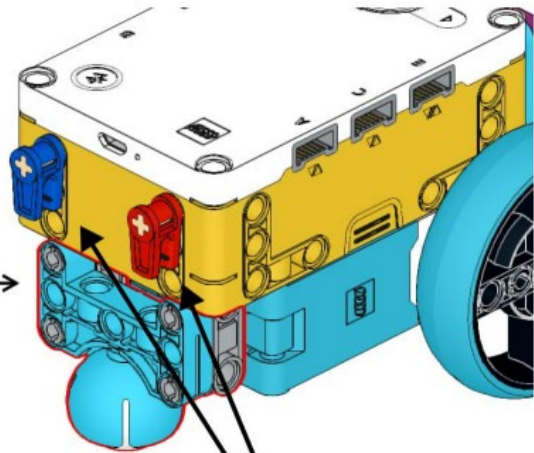
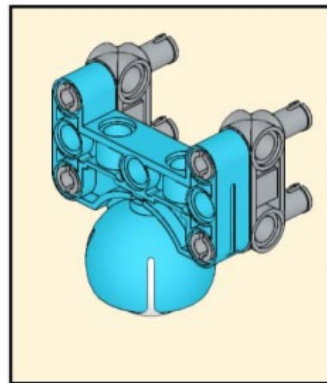
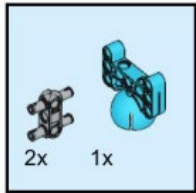
Kabelführung



unter Hub



11



Kabelführung

Farbsensor

Clips für die beiden Motorenkabel

Motor links

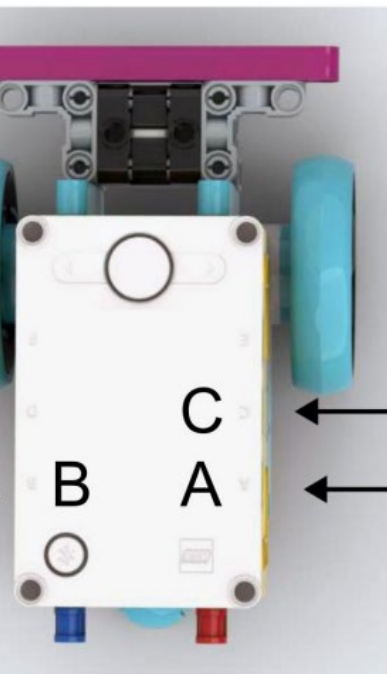
Motor rechts



phobricks

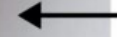
Anschlüsse

Motor links



Farbsensor

Motor rechts



phobricks



7: Facciamo muovere il nostro robot

Con i primi passi della sezione «Avvio» hai messo in funzione un motore. Se però il robot deve muoversi come un'auto, devono funzionare contemporaneamente due motori.

7.1 Inizia un nuovo progetto

Obiettivo di apprendimento:

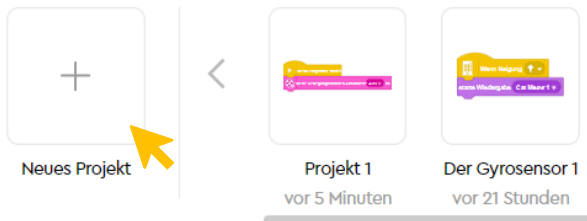
- So come iniziare un nuovo progetto nell'app SPIKE.

Vai sulla **Pagina iniziale** dell'app SPIKE. La puoi raggiungere da qualsiasi sezione dell'app, con il simbolo della casetta.

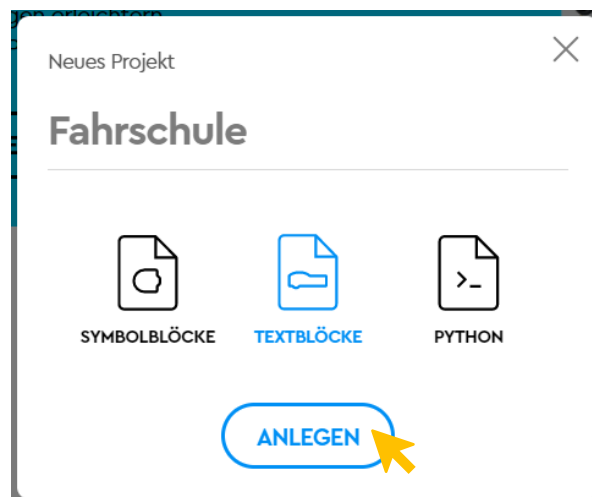


Quindi clicca su **Crea un progetto** (sotto l'area blu con Avvio).

Zuletzt verwendete Projekte



A questo punto si aprirà una piccola finestra con la scritta «Progetto 1» in grigio chiaro nella parte superiore. Qui devi inserire un nome adatto per il tuo progetto, in modo da poterlo poi ritrovare facilmente: ad esempio «Scuola guida». Quindi seleziona **Blocchi di testo** e clicca su **Crea**.



7.2 Impostare i motori di movimento

Obiettivo di apprendimento:

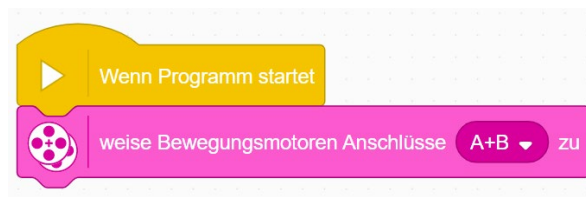
□ So cosa sono i motori di movimento e come impostarli per un progetto.

Innanzitutto dobbiamo insegnare al programma quali motori servono al robot per muoversi, qual è quello di destra e quale quello di sinistra. Questo è ciò che si intende con **motori di movimento**.




Consiglio

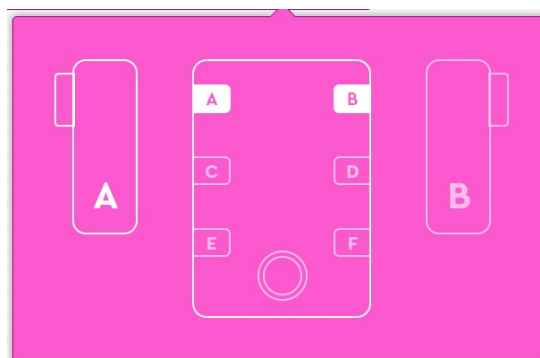
Se impostiamo subito all'inizio i motori di movimento, il programma saprà sempre cosa fare e non sarà necessario dirglielo ogni volta. Ciò è importante soprattutto nei programmi più lunghi, in cui si ripetono continuamente operazioni come la marcia, lo stop o il cambio di direzione.



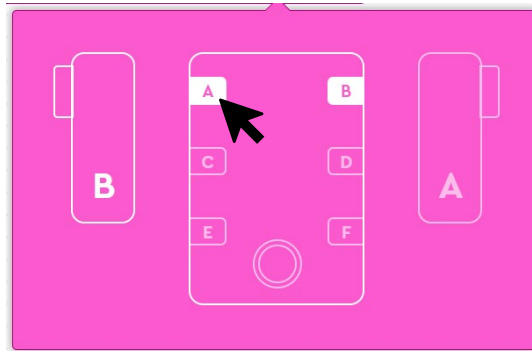
Questa è l'impostazione predefinita di questo blocco ed è perfetta se il tuo **motore sinistro** è collegato alla porta **A** e quello destro alla porta **B**. Logicamente, se i tuoi motori non sono collegati alle porte A e B, bisogna selezionare altre lettere.

Nel caso in cui dopo l'avvio del programma il robot si muova nella direzione sbagliata (ossia all'indietro invece che in avanti), devi modificare l'assegnazione dei motori di movimento. Naturalmente potresti anche semplicemente scollegare e ricollegare i cavi in modo che A sia a sinistra e B a destra. Ma in base al tipo di struttura del tuo robot, ciò potrebbe non essere possibile. Per questo puoi quindi modificare l'assegnazione anche attraverso il codice. Procedi come illustrato qui sotto.

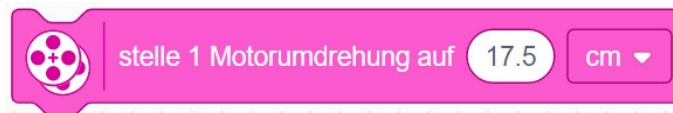
Clicca sul campo delle impostazioni .
Comparirà questa finestra:



Qui puoi vedere a quali due porte sono collegati i motori, e qual è il motore di sinistra. Se clicchi su una delle due porte, l'assegnazione si inverte: adesso **B** è a **sinistra** e **A** a **destra**.



Possiamo anche impostare subito le dimensioni delle ruote (circonferenza della ruota), visto che c'è ovviamente una differenza se ci si muove con ruote grandi o ruote piccole. Con una rotazione il robot fra più strada se le ruote sono grandi che non se sono piccole. Le ruote turchese in dotazione nello SPIKE-Prime-Box sono grandi 17,5 cm, il che corrisponde all'impostazione predefinita, se mai dovessi dimenticarlo. Ma anche per una questione stilistica è sempre meglio inserire le dimensioni delle ruote.



Infine impostiamo la velocità standard. Verrà sempre utilizzata questa velocità, se non diversamente indicato. Le velocità vengono sempre indicate in percentuale. 0 % significa nessun movimento, mentre 100 % è la velocità massima del motore. Se scrivi un numero superiore a 100, il robot lo interpreterà automaticamente come se fosse 100. Non è possibile forzare il robot ad andare più veloce, a meno che non utilizzi degli ingranaggi; ma questo è un altro argomento di cui qui non parleremo.



Consiglio

Per gli esercizi è meglio scegliere una velocità bassa. Nella maggior parte dei casi il robot risulta così un po' più preciso. Inoltre si può osservare meglio cosa succede e riconoscere eventuali errori. E poi c'è sempre tempo per andare più veloci!

Al termine dell'assegnazione dei motori di movimento, il tuo codice apparirà come illustrato qui di seguito.



Questi tre blocchi di movimento devono essere inseriti all'inizio di ogni programma in cui il tuo robot debba spostarsi da qualche parte. Naturalmente devi sempre selezionare i motori giusti ed impostare correttamente la circonferenza delle ruote e la velocità.

7.3 Marcia semplice

Obiettivo di apprendimento:

- So far partire il mio robot e farlo fermare con un blocco «Attendi»

Ora siamo finalmente pronti per mettere in marcia il nostro robot!

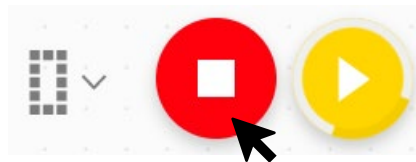
Sotto l'impostazione dei motori di movimento c'è un **blocco «Avvio»**: la freccia verso l'alto indica che il robot deve procedere verso avanti.



Blocco «Avvio»

Cosa succede? E adesso come puoi fermare nuovamente il robot? Di sicuro non vuoi farlo andare avanti fino a quando si scarica la batteria!

Per fermarlo premi il pulsante rotondo sul robot, o clicca il pulsante rosso Stop nell'app SPIKE.



Ma noi vogliamo che il robot si fermi da solo!

Ora vogliamo che il robot si muova esattamente per 1 secondo prima di fermarsi nuovamente. Per questo ci serve un **blocco Attendi** arancione e un **blocco Arresta movimento**.



Blocco «Avvia movimento»

Blocco «Attendi»

Blocco «Arresta movimento»

7.4 Fermarsi sulla linea nera con un blocco «Attendi fino a quando»

Obiettivi di apprendimento:

- So cos'è un blocco «se-allora» e so utilizzarlo per far fermare il mio robot.
- So utilizzare il sensore di colore.

Adesso vogliamo far andare il robot non per un solo secondo, ma fino a quando non arriva ad una linea nera.

Ci serve quindi invece che un semplice blocco Attendi un **blocco Attendi fino a quando**



Blocco «Attendi fino a quando»

Nell'esagono deve essere inserita la condizione «arrivare ad una linea nera». Per questo ci serve un **sensore di colore**. Andiamo quindi nei blocchi di sensore azzurri e ne cerchiamo uno con cui riconoscere il colore. Fai attenzione a selezionare la porta giusta (**C**) per il sensore di colore!



Consiglio

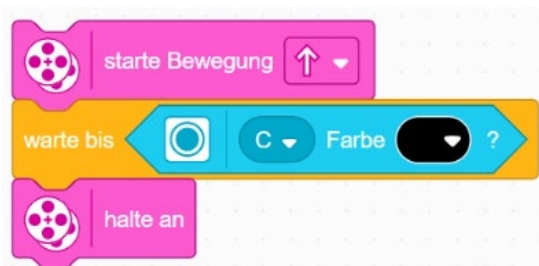
Per tutti i sensori ci sono blocchi ovali che ti indicano sempre il valore misurato attuale e blocchi esagonali che confrontano il valore misurato attuale con quello desiderato. Nel blocco «Attendi fino a quando» ci serve sempre il blocco esagonale!

Ci serve il blocco turchese «**il colore è?**».



Ma attenzione! Con questa impostazione il sensore cerca non il colore nero, come ci serve per l'esercizio, ma il rosso! Clicca quindi sulla piccola freccia nel campo del colore e seleziona il nero. Inserisci quindi questo blocco nel campo esagonale «**Attendi fino a quando**».

Il codice completo (dopo l'impostazione dei motori di movimento) apparirà come illustrato qui di seguito.



Blocco «il colore è?» in un blocco «Attendi fino a quando»

Esercizio

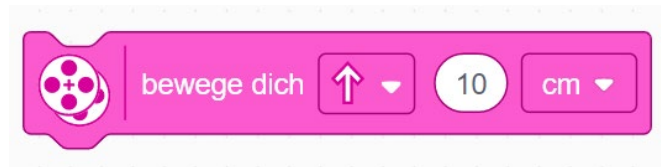
Il tuo robot non deve solo fermarsi ma anche far comparire una «X» nella matrice luce e riprodurre un segnale acustico. Come completeresti il codice?

7.5 Blocco «Muoviti per una durata determinata»

Obiettivo di apprendimento:

- Conosco il blocco «Muoviti per una durata determinata».

Il blocco Avvio è quindi utile quando non sappiamo di preciso quanto lontano deve andare il robot. Spesso però lo sappiamo, poiché abbiamo a disposizione un righello e conosciamo esattamente il percorso. Possiamo quindi prendere anche il **blocco «Muoviti per una durata determinata»**.



Esercizi

1. Come puoi far andare il tuo robot in retromarcia?
2. Fai andare avanti il tuo robot esattamente per 29 cm. Controlla con un foglio A4 se la distanza coincide!
3. Fai andare avanti il tuo robot esattamente per una rotazione. A quanti centimetri corrisponde?
4. Fai andare avanti il tuo robot con una velocità del 20 % esattamente per 3 secondi accanto ad un righello. Successivamente modifica la velocità standard dei motori di movimento portandola al 40 % e fai nuovamente andare avanti il robot per 3 secondi. Cosa noti?
5. Fai fare al tuo robot una rotazione su se stesso. Quanti centimetri devi inserire affinché alla fine si trovi esattamente nel punto iniziale?

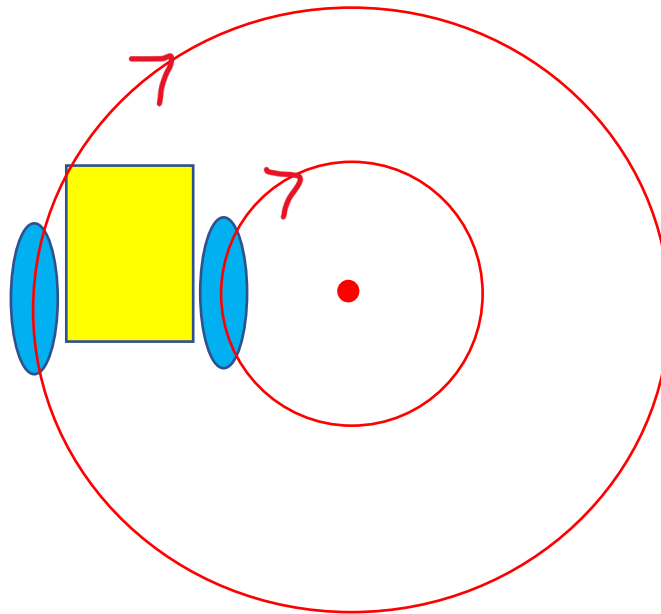
7.6 Ruotare e svoltare

Obiettivo di apprendimento:

- Conosco diverse possibilità per far svoltare il mio robot.

Finora il tuo robot poteva andare dritto, o al massimo lo hai fatto andare in retromarcia. In realtà può anche ruotare e svoltare. Per questo ci sono diverse possibilità.

1. **Curve:** puoi fargli fare una curva (che in realtà è una parte di un cerchio).



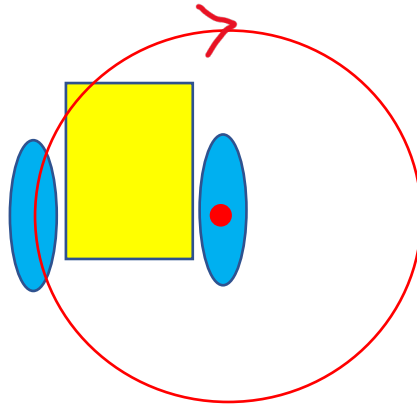
Per questo ci serve il **blocco «Dirigiti per una durata limitata»:**



Esercizio

Prova a vedere cosa succede se aumenti o diminuisci il numero nel primo campo di impostazione!

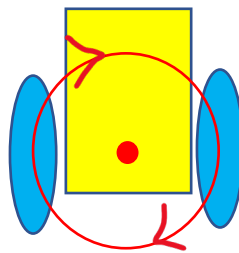
2. **Rotazione circolare:** una forma speciale di curva è la «rotazione circolare». Funziona esattamente come un cerchio in geometria. Una ruota rimane ferma, mentre l'altra si muove.



Per questo tipo di rotazione ci serve sempre lo stesso blocco **«Dirigiti per una durata limitata»**. Adesso però dobbiamo impostare il valore del campo destra/sinistra esattamente a **50** se vogliamo ruotare verso destra e a **-50** se vogliamo ruotare verso sinistra.



3. **Rotazione su se stesso:** con la rotazione su se stesso il robot ruota sul punto centrale fra le sue due ruote motrici.



Questa è la rotazione più facile: scegli semplicemente la **freccia curva** nel **blocco «Muoviti per una durata determinata»**



Esercizio

Quanti centimetri ti servono per una rotazione su se stesso completa del robot?

Se hai un po' di tempo per provare, modifica il tuo robot rendendolo più largo. E adesso scopri nuovamente quanti centimetri ti servono per una rotazione su se stesso del robot.

7.7 Condizioni

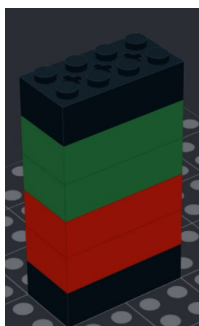
Obiettivi di apprendimento:

- Posso modificare il mio robot in modo da renderlo adatto ad altri compiti.
- So come programmare usando le condizioni «se-allora» e «se-allora-altrimenti».

Adesso vogliamo ritornare nuovamente alla nostra auto robot che si muove da sola. Ti ricordi? Quando arriva alla linea gialla, deve controllare il colore del semaforo e comportarsi di conseguenza. Adesso vogliamo programmarla con precisione.



Purtroppo nello SPIKE Prime-Box c'è solo un sensore di colore, e quindi dobbiamo utilizzare quello anche per riconoscere il semaforo. Il robot deve quindi essere modificato in modo da guardare in avanti invece che in basso. Prova ad essere creativo, non è difficile! Quindi devi anche costruire uno o due oggetti da utilizzare come semaforo. Il colore deve trovarsi alla stessa altezza del sensore di colore! Questa immagine può darti un suggerimento, ma in base al modo in cui hai messo il sensore, il tuo oggetto potrebbe anche essere totalmente diverso dal nostro.




Ora metti il robot su un tavolo, stando attento che non ci sia niente davanti al sensore.

Controlla se l'hub è acceso e collegato.

Adesso controlla cosa vede il tuo sensore.



Puoi trovare questa piccola grafica informativa in alto a sinistra nel tuo progetto. Qui puoi vedere a quali porte sono collegati i tuoi motori (**A** e **B**) e in quale posizione si trovano esattamente. Adesso però a noi interessa cosa ci dice il sensore di colore collegato alla porta C. Non riconosce alcun colore. Questo è ciò che significa questo segno: 

Adesso posiziona il tuo «semaforo» davanti al sensore, all'incirca alla distanza di un dito. Ora il sensore di colore collegato alla porta C dovrebbe mostrare il colore giusto.

Facciamo partire il nostro robot come sempre con un blocco «**Avvia movimento**» ed un blocco «**Attendi fino a quando**».

Ora dobbiamo ancora differenziare fra quando non c'è nessun colore e quando il semaforo è verde o rosso.

Innanzitutto vogliamo controllare se viene riconosciuto un qualsiasi colore. Per farlo utilizziamo il blocco «**il colore è?**».



Però ci serve un colore! Purtroppo un blocco del tipo «Il sensore vede un qualsiasi colore?» non esiste. Per questo ci serve un blocco che trasformi questo blocco di sensore nel suo opposto. Ci serve cioè il blocco «**Non**».



Come puoi vedere, il blocco esagonale ha spazio per un altro blocco esagonale; quindi trasciniamo semplicemente il nostro blocco turchese «**il colore è?**» nel blocco «**Non**».



E quindi li spostiamo tutti e due assieme in un blocco «**Attendi fino a quando**».



Il robot si muove così fino a quando il sensore scopre l'oggetto-semaforo.

A quel punto deve decidere. Fermarsi o proseguire?

Per la decisione ci serve ancora un blocco C, ossia il blocco «se-allora».



Ora rifletti: cos'è più importante? Che il robot scopra che il semaforo è verde o che scopra che è rosso? Nel traffico stradale non è tanto grave se non vedi un semaforo verde. Passi comunque senza creare problemi. Ma se non vedi un semaforo rosso, allora può essere pericoloso. Quindi il robot deve cercare il semaforo rosso.



Nota

Cerca sempre e innanzitutto ciò che è più importante!

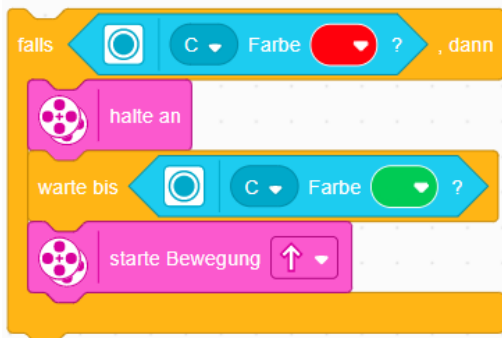
Per questo inseriamo il blocco «**il colore è?**» con il colore che cerchiamo nell'esagono del blocco «**se-allora**».

E nel blocco C scriviamo cosa dobbiamo fare se questa condizione si verifica.



Con questo codice il robot si ferma solo quando il colore che ha trovato è rosso. Se non è rosso il robot continua con ciò che si trova al di sotto del blocco C.

Possiamo modificare il codice in modo che il robot prosegua quando vede verde.

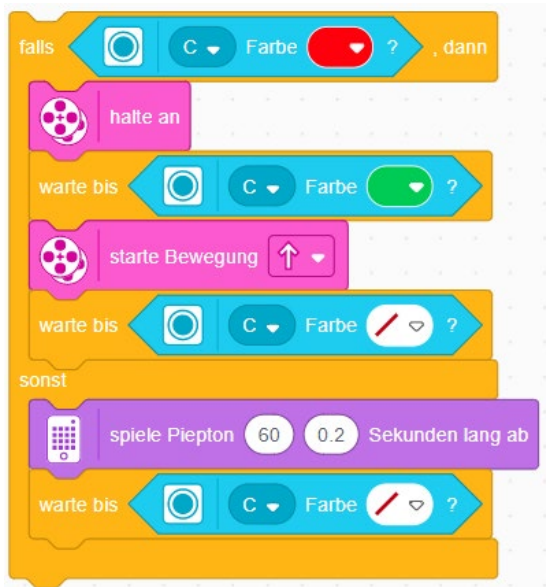


Se il robot deve fare qualcosa di speciale quando vede verde, ci serve il blocco a forma di E «**se-allora-altimenti**» invece di quello a forma di C «**se-allora**».



Nel nostro esempio il robot deve emettere un breve segnale acustico, in modo che noi capiamo che ha riconosciuto il semaforo verde; siccome si sta già muovendo non ci serve però un nuovo blocco «**Avvia movimento**»!

Il nostro codice appare quindi come illustrato qui di seguito.

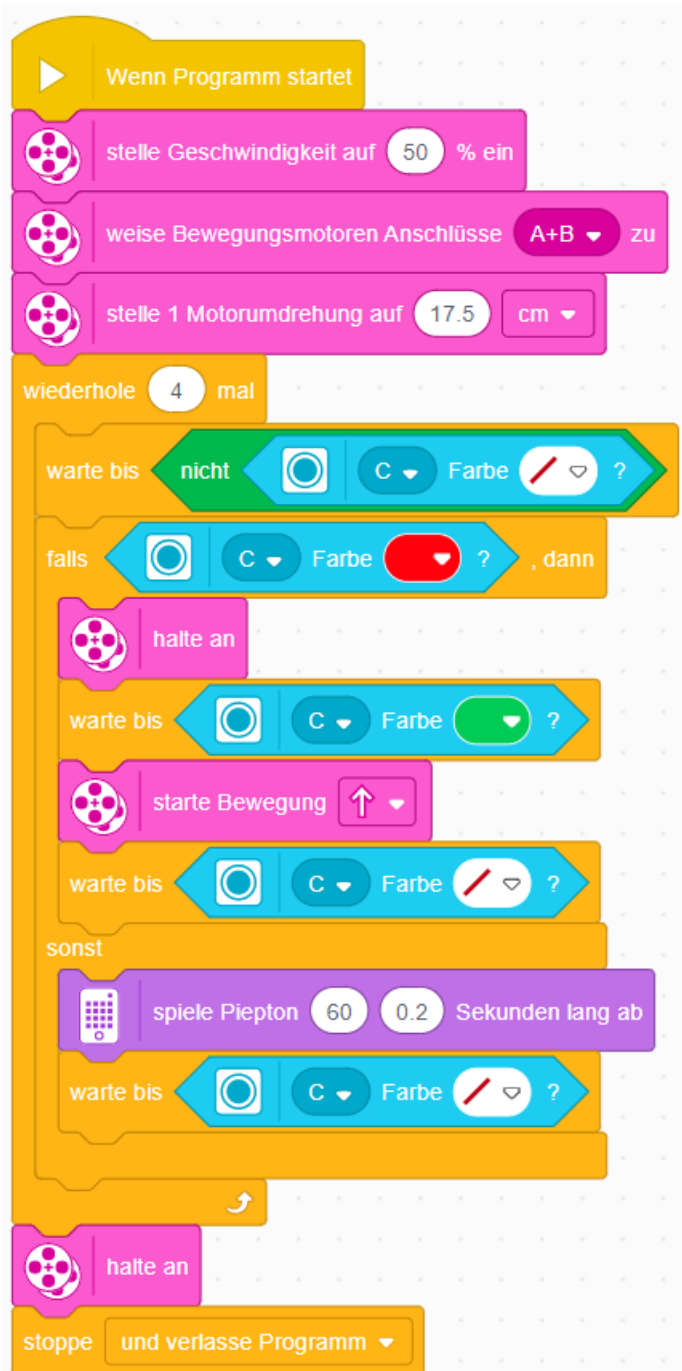


I due blocchi con cui si attende finché non c'è più un colore da vedere sono un trucco per guadagnare tempo per riorganizzare i blocchi semaforo.

Se sappiamo che sul nostro percorso ci sono 4 semafori uno dopo l'altro, possiamo inserire nel blocco E anche un ciclo. Alla fine del ciclo devi comunque fermare il robot e terminare il programma, per evitare che vada avanti all'infinito!



Il nostro codice completo appare quindi come illustrato qui di seguito.



The image shows a Scratch-style block diagram for a robot program. The blocks are as follows:

- Start:** A yellow block labeled "Wenn Programm startet" (When program starts).
- Motor Speed:** A pink block "stelle Geschwindigkeit auf 50 % ein" (set speed to 50 %).
- Motor Connections:** A pink block "weise Bewegungsmotoren Anschlüsse A+B zu" (assign motor connections A+B).
- Motor Distance:** A pink block "stelle 1 Motorumdrehung auf 17.5 cm" (set 1 motor rotation to 17.5 cm).
- Loop:** An orange "wiederhole 4 mal" (repeat 4 times) block containing:
 - A green "warte bis nicht" (wait until not) block with a sensor icon, port "C", and "Farbe" (color) set to a red line.
 - An orange "falls" (if) block with a sensor icon, port "C", and "Farbe" (color) set to a red circle. Inside:
 - A pink "halte an" (stop) block.
 - A blue "warte bis" (wait until) block with a sensor icon, port "C", and "Farbe" (color) set to a green circle.
 - A pink "starte Bewegung" (start movement) block with an upward arrow.
 - A blue "warte bis" (wait until) block with a sensor icon, port "C", and "Farbe" (color) set to a red line.
 - An orange "sonst" (else) block containing:
 - A purple "spiele Piepton 60 0.2 Sekunden lang ab" (play beep tone 60 0.2 seconds long) block.
 - A blue "warte bis" (wait until) block with a sensor icon, port "C", and "Farbe" (color) set to a red line.
- End:** A pink "halte an" (stop) block and a yellow "stoppe und verlasse Programm" (stop and leave program) block.

Parte 4: Algoritmi importanti

Nota

Un algoritmo è un compito che in un programma si ripete di continuo e funziona autonomamente.

8: «Rimani nell'arena»

Obiettivi di apprendimento:

- Conosco il termine «algoritmo».
- So programmare il «Gioco dell'arena»
- Per il livello avanzato: so espandere con altri eventi il «Gioco dell'arena».

In questo esercizio userai tutto quello che hai imparato finora. Luci, suoni, movimenti, cicli e condizioni. Cicli e condizioni possono anche essere inseriti uno dentro l'altro.

Esercizio

Crea un nuovo progetto con il nome «Gioco dell'arena».

Il tuo robot deve restare sempre all'interno della linea nera. Quando arriva alla linea deve:

- fermarsi
- emettere un breve segnale acustico
- mostrare per 0,2 secondi una freccia verso il basso sulla matrice luce
- andare in retromarcia per 8 centimetri facendo un ampio arco
- riprendere a proseguire in avanti

Il tutto deve ripetersi 10 volte. Dopo la decima volta il robot deve fermarsi ed applaudire.



Consiglio

All'inizio prova a svolgere l'esercizio da solo! Prova sempre se tutto funziona come vuoi tu e modifica opportunamente il codice affinché ciò accada. Se incontri delle difficoltà, ci sono le seguenti possibilità.

- Cerca innanzitutto di programmare cosa deve fare il robot quando arriva alla linea nera. Preoccupati solo in seguito di inserire tutto questo in una condizione.
- Se incontri delle difficoltà con la struttura del codice, allora guarda a pagina 26 (nel dossier per gli insegnanti a pagina 33), in fondo a questo dossier. Lì troverai il codice, però completamente privo di scritte. In base ai colori e alle forme puoi desumere come dovrebbe essere.
- Se anche questo non ti aiuta, allora chiedi supporto al direttore del tuo corso o all'insegnante.

9: Segui la linea!

Obiettivi di apprendimento:

- So per cosa è utile un inseguitore di linea.
- So programmare un semplice inseguitore di linea.

Alle volte puoi non sapere in anticipo quale percorso debba esattamente seguire il tuo robot. O il percorso è talmente complicato che non puoi programmarlo semplicemente attraverso i diversi blocchi di movimento. Per questo c'è quindi un prezioso aiuto: delle linee nere che il tuo robot può seguire!

Esercizio

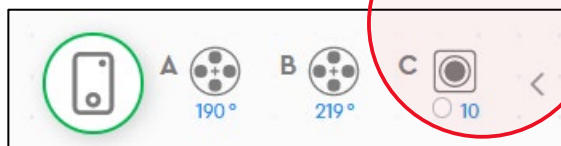
Sul pavimento del corridoio c'è una lunga linea di nastro adesivo. Tu ed il tuo compagno/compagna formate una squadra. Un membro della squadra è il «robot» e sta in piedi con gli occhi bendati all'inizio della linea. L'altro membro della squadra è il «programmatore/programmatrice» e gli fornisce le istruzioni per seguire la linea. Quindi scambiatevi i ruoli.

Osserva come il «robot» si muove lungo la linea!

Ora vogliamo programmare un inseguitore di linea che funzioni in maniera del tutto analoga.

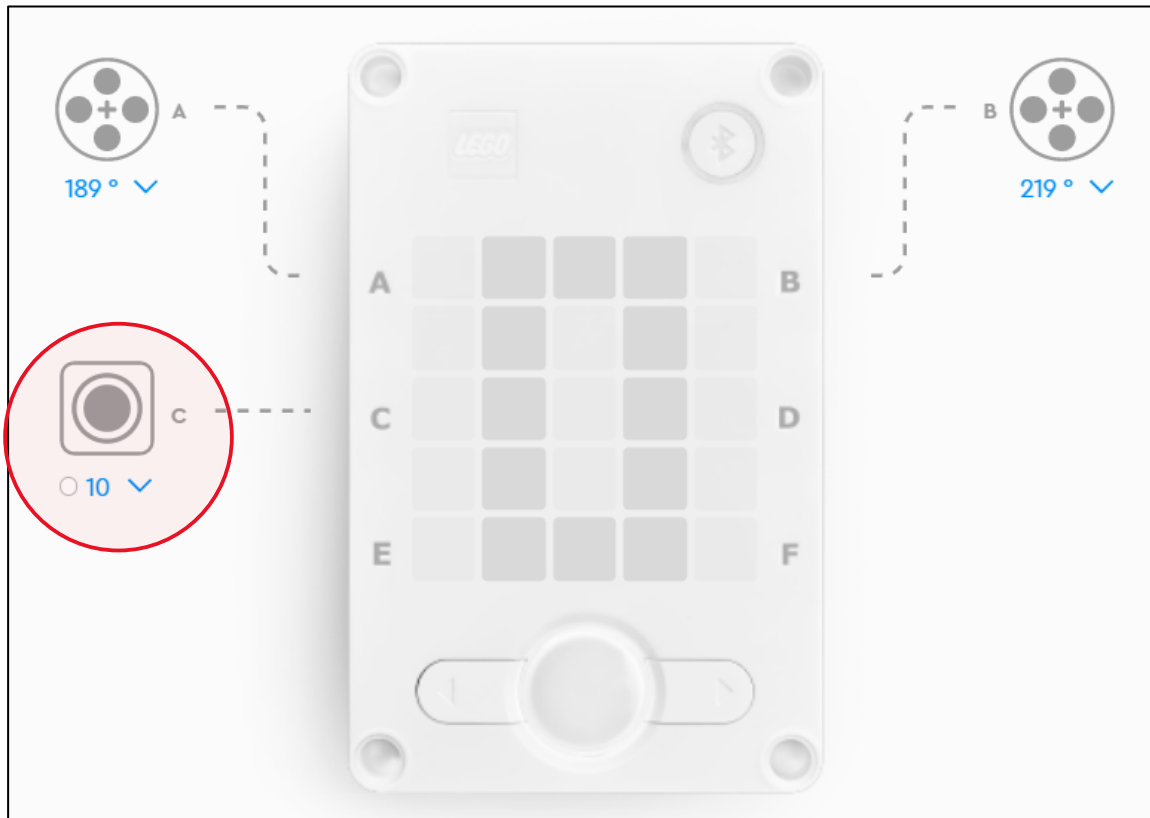
Prima però dobbiamo provare ancora una cosa. Per questo ti serve nuovamente il **sensore di colore** sulla parte anteriore del robot.

Posiziona il tuo robot con il sensore di colore su una superficie bianca, ad esempio su un foglio di carta. Adesso guarda sull'app SPIKE quale colore viene mostrato. Dovrebbe essere bianco.



Il sensore mostra bianco. «10» è il numero che in SPIKE indica il bianco.

Adesso clicca sul simbolo dell'hub. In questo modo ottieni una vista più ampia dell'hub, con i motori e i sensori collegati.



Se clicchi sulla piccola freccia sotto il simbolo del sensore di colore, si apre un menu di selezione. Seleziona «Rifletti».



Ora se chiudi la vista ampia dell'hub, il display di controllo dovrebbe apparire come illustrato di seguito.

SWITZERLAND



Il valore è molto alto, quasi il 100 %. Se il valore è inferiore può dipendere da un'illuminazione insufficiente o dal fatto che la superficie non sia completamente bianca. Oppure il tuo sensore è stato montato troppo in alto. Non è grave.

Ripeti la stessa procedura per il nero. Il valore è molto basso, nella maggior parte dei casi inferiore al 20 %.

Posiziona quindi il tuo robot in modo che il sensore di colore si trovi esattamente sopra il bordo destro (in direzione di marcia) della linea, e misura il valore del sensore. Dovrebbe essere compreso fra il 40 % e il 60 %. Questo è il nostro **valore di soglia**.



✗ Completamente sopra la linea nera. Valore del sensore troppo basso.



✗ Troppo sulla zona bianca. Valore del sensore troppo alto.



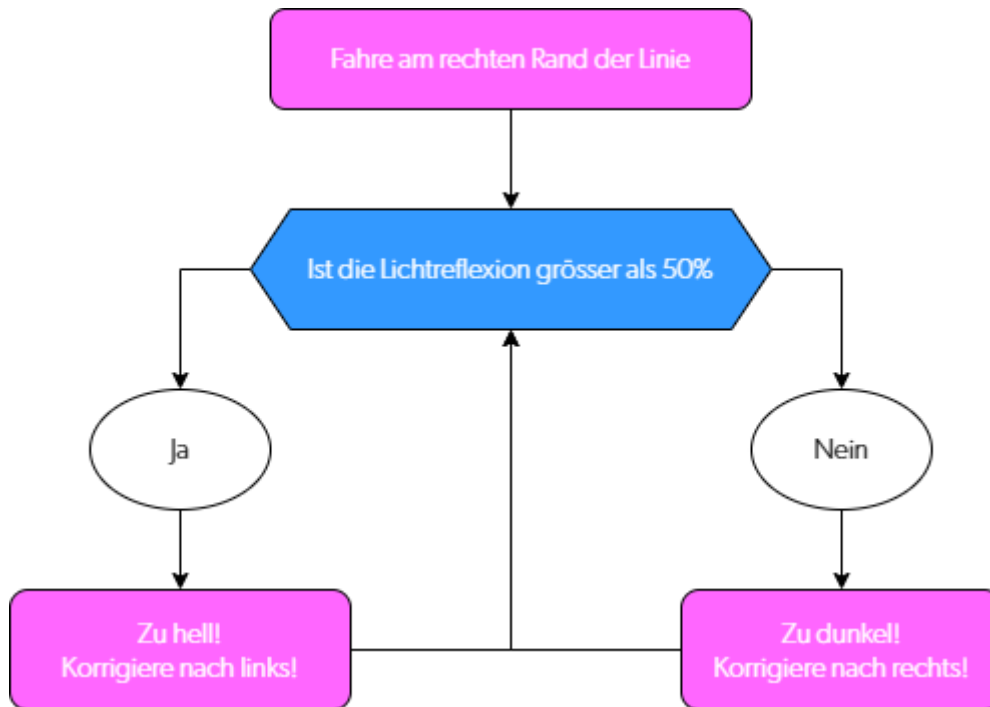
Esatto!

Adesso ricorda l'esercizio iniziale sulla linea nera: quando tendevi ad andare a destra, dovevi correggere verso sinistra, mentre quando tendevi ad andare a sinistra, dovevi correggere verso destra.

Per il nostro robot adesso vale la stessa identica cosa. Quando il robot tende ad andare a destra, il sensore di colore diventa troppo chiaro (quindi il numero diventa troppo grande) e deve essere corretto verso sinistra, mentre quando il robot tende ad andare a sinistra, il valore soglia diventa troppo piccolo, quindi troppo scuro, e deve essere corretto verso destra.

Tutto questo deve ripetersi all'infinito (o fino a quando si termina il programma).

Per questo esercizio vogliamo seguire la linea fino a quando il robot non arriva ad una distanza inferiore ai 15 cm da un ostacolo. Per questo dobbiamo montare nella parte anteriore del robot anche un **sensore di distanza**.



Il codice appare come segue.

Blocco «se-allora-altrimenti»

Condizione: riflesso della luce inferiore al 50 % (o inferiore al valore che hai misurato sul bordo)

Allora (vero): troppo scuro → correzione verso destra

Altrimenti (falso): troppo chiaro → correzione verso sinistra

Osserva: come si muove adesso il robot? Come si potrebbe chiamare questo tipo di inseguitore di linea?

Esercizio

Prova a vedere cosa succede se utilizzi un valore di correzione maggiore o minore. Prova anche a vedere se cambiando i valori di correzione c'è una differenza nella velocità con cui funziona il robot.

Aiuto per la soluzione dell'esercizio «Resta nell'arena»

Pensa a questo: all'inizio del codice devi impostare i motori di movimento. Nel blocco «Attendi fino a quando» stabilisci cosa deve fare il robot quando arriva ad una linea nera, che riconosce attraverso il sensore di colore.

Attenzione: se vuoi che il robot percorra un ampio arco in retromarcia, devi inserire prima dell'unità di misura (cm o rotazioni) un numero negativo (un numero con un meno davanti), in modo che il robot si muova in direzione opposta.

