

Presentazione mbot: programmazione a blocchi per il line following

Inizio per classi che non fanno il SIA:

Ciao a tutti! Siamo la terza SIA, e oggi vi mostreremo qualcosa che abbiamo costruito e programmato noi stessi.

Visto che non frequentate l'indirizzo SIA prima però vogliamo raccontarvi come ci siamo arrivati, perché non è che il primo giorno di scuola ci hanno messo davanti a un robot e ci hanno detto "programmate". C'è stato un percorso.

Tutto è partito da Flowgorithm, un software che vi permette di creare algoritmi visivamente, con frecce e blocchi.

Un algoritmo, in parole semplici, è una sequenza di istruzioni che porta a un risultato, come una ricetta di cucina, ma per il computer.

Flowgorithm ci ha insegnato a pensare prima di scrivere: capire il problema, scomporlo in passi, e solo poi tradurlo in codice.

Da lì siamo passati alla programmazione a blocchi quella che vedete sullo schermo, quella che usa anche il robot qui davanti a voi.

È un linguaggio visivo: non scrivi testo, trascini pezzi di codice e li incastri come un puzzle. Sembra semplice, ma vi assicuro che anche qui bisogna ragionare bene su cosa stai facendo.

Infine siamo arrivati a Java, un linguaggio di programmazione vero e proprio, usato professionalmente in tutto il mondo.

Lì le cose si complicano: variabili, classi, oggetti, metodi ma è anche dove capisci davvero come funziona un programma dall'interno.

Questo percorso, dall'algoritmo su carta fino al robot che si muove da solo, è quello che fa il nostro indirizzo.

Non si impara solo a usare il computer si impara a far fare al computer quello che vuoi tu..

Questo è un programma di line following, il robot segue autonomamente una linea nera su sfondo umano

Inizio per classi SIA:

Salve a tutti. siamo la terza SIA, e quello che vedete qui, come probabilmente avete già notato, è un programma programmato a blocchi di line following un robot che segue autonomamente una linea nera su sfondo bianco, senza intervento umano.

Presentazione sul funzionamento:

Voglio spiegarvi come funziona, ma prima voglio sottolineare i concetti fondamentali di programmazione che ci sono dietro, perché questo piccolo programma li racchiude quasi tutti.

Il primo è l'iterazione. Il programma gira in un ciclo infinito quel per sempre in alto si ripete decine di volte al secondo.

Senza questa ripetizione continua, il robot valuterrebbe la situazione una volta sola e si fermerebbe.

L'iterazione è il cuore pulsante di qualsiasi sistema in tempo reale.

Il secondo è la selezione, ovvero la struttura se–altrimenti.

Il robot non ha occhi, ha un sensore di inseguimento linea sulla porta 2, che rileva se sotto di lui c'è bianco o nero, lato sinistro e lato destro. In base a quella lettura, il programma sceglie cosa fare.

Questa è esattamente la logica condizionale che trovate in qualsiasi linguaggio di programmazione, da Python a Java a C++.

Il terzo concetto, meno ovvio ma presente, è quello di input e output. I LED colorati non sono decorativi: sono un feedback visivo dello stato interno del robot. Blu significa che sta correggendo verso destra, rosso verso sinistra. Questo è un principio fondamentale nel debug e nel monitoraggio di sistemi embedded.

Ora, la logica concreta:

Se entrambi i lati vedono bianco → il robot è sulla linea, va dritto al 40%

Se il lato destro vede nero → il robot ha svicolato a destra, quindi la ruota destra rallenta al 30% e quella sinistra accelera al 70%, correggendo la traiettoria verso sinistra

Se il lato sinistro vede nero → vale il contrario: ruota sinistra al 30%, destra al 70%

È un sistema di correzione continua, un principio che ritrovate per esempio nei sistemi di guida autonoma, in qualsiasi macchina che deve mantenere una direzione.

Tutto questo, scritto in poche righe di logica condizionale all'interno di un loop.