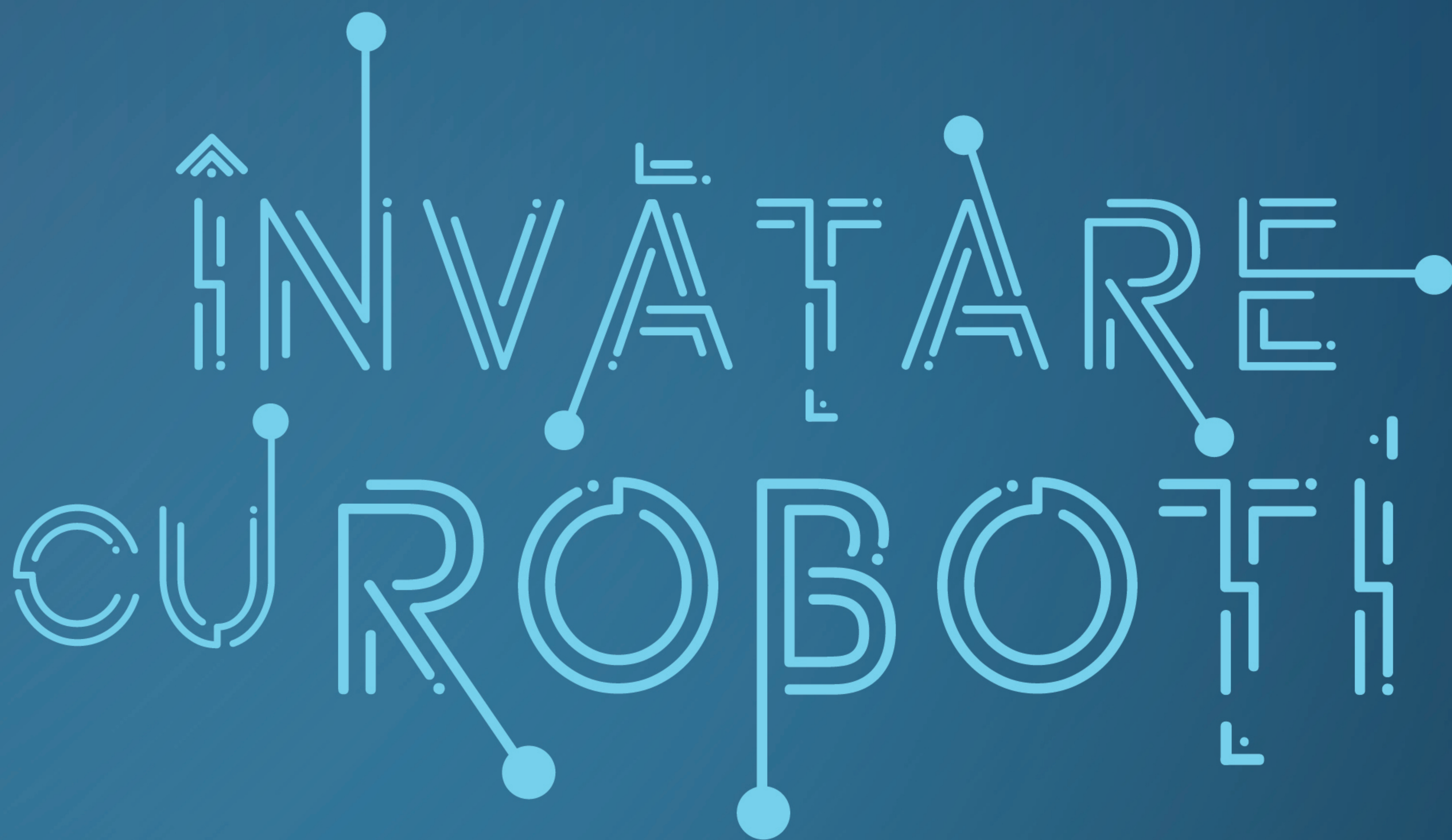




Google.org



ECIVIS



INVĂȚĂRE CU ROBOTI

SUPPORT DE CURS
MARILENA OPREA





ÎNVĂȚARE CU ROBOȚI

SUPPORT DE CURS

PROF. MARILENA OPREA

”Învățare cu roboți” este un suport de curs dedicat în special profesorilor din România care predau disciplina Informatică și TIC la clasele gimnaziale, dar care poate fi folosit de oricine dorește să învețe sau să predea noțiuni de robotică.
Acest suport de curs este gratuit.

Autor: prof. Marilena Oprea
Editor: Ana-Maria Stancu
Asociația E-Civis, 2018

Proiectul ”Învățare cu roboți” este finanțat prin programul **Google Educator Grants**.



CUPRINS

DESPRE PROIECT	5
SCOPUL PROIECTULUI	5
OBIECTIVE DE ÎNVĂȚARE	5
SUPPORTUL DE CURS	6
I.PLATFORMA OPEN ROBERTA LAB	11
Meniurile platformei OPEN ROBERTA LAB	12
Simulatorul OPEN ROBERTA SIM	16
II. LIMBAJUL NEPO	21
Categoriile de blocuri	24
Categoria Acțiune (Action)	24
Categoria Control	28
Categoria Logic	31
Categoria Matematică (Math)	32
Categoria Text	35
Categoria Variabile (Variables)	36
Categoria Liste (List)	37
Categoria Funcții (Functions)	38
Categoria Mesaje (Messages)	39
Categoria Senzori (Sensors)	39
Categoria Culori (Colors)	40
III.LECȚII	43
Lecția Urmărirea liniei	44
Lecția Senzorul tactil	46
Lecția Senzorul ultrasonic	47
IV. MATERIALE DIDACTICE	51
V. ROBOTICA COMPETIȚIONALĂ	59
Competiții de robotică organizate în România	60
Despre Asociația E-Civis	63
Despre autor	64
INDEX	67





Despre Proiect

Proiectul "Învățare cu roboți" propune o modalitate de predare a lecțiilor de robotică incluse în curriculumul național pentru clasele a VII-a și a VIII-a, ca urmare a schimbării planurilor cadru și a programelor școlare la gimnaziu prin OMEN nr. 3393 / 28.02.2017.

Anul școlar 2017-2018 a adus schimbări majore în curriculum școlar românesc prin introducerea unei discipline noi la gimnaziu și anume Informatică și TIC, a orelor de programare și robotică la clasele VII-VIII. Este o inițiativă foarte bună pe care doar câteva state din Europa au implementat-o până în prezent, dar prezintă și o provocare pentru sistemul educațional din România. Aspectul cel mai important în producerea acestei schimbări îl constituie resursa umană, deoarece profesorii de gimnaziu din România, în mare parte, nu sunt foarte familiarizați cu programarea și, mai ales, cu robotica.

În acest context, pentru a veni în sprijinul profesorilor care predau Informatică și TIC la gimnaziu, Asociația E-Civis implementează proiectul Învățare cu roboți care cuprinde realizarea acestui suport de curs precum și organizarea unor întâlniri demonstrative în diferite regiuni ale țării, în principal cu profesorii care predau la gimnaziu Informatică și TIC, cu profesorii de informatică, dar și cu cei interesați de educația STEAM, pentru a se prezenta un model viabil de predare a roboticii educaționale la clasă.

Rezultatele proiectului vor fi: 150 de profesori instruiți, un suport de curs, o rețea de profesori, materiale didactice care pot fi folosite de profesori în școli, dar și de cluburile de robotică publice sau private.

Scopul proiectului

Promovarea roboticii în mediul educațional

Obiective proiectului

- înțelegerea conceptelor de bază în robotică: senzori, intrări, ieșiri etc.
- înțelegerea unui mediu de simulare și de testare a activității unui robot
- înțelegerea unui mediu de programare
- inițierea unor activități de învățare cu roboți
- diseminarea informațiilor de bază despre industria roboticii și despre potențialul acesteia





Suportul de curs

Cursul este structurat în cinci capitole, primele trei prezentând conținuturi selectate astfel încât să fie atinse competențele prevăzute în programa școlară aprobată prin OMEN 3393 / 28.02.2017. Cunoștințele noi sunt detaliate în exemple, activități de învățare ce cuprind sugestii de rezolvare și soluții, teme de discuții, probleme propuse.

În primul capitol sunt prezentate elementele specifice platformei OPEN ROBERTA LAB (meniurile, setările, simulatorul, tipurile de roboți) și elementele de sprijin precum comunitatea celor care folosesc platforma. Capitolul al II-lea prezintă limbajul de programare NEPO, fiecare categorie de blocuri fiind urmată de modele de activități de învățare și aplicații propuse. Deși platforma permite programarea a 7 tipuri de roboți, suportul de curs exemplifică programarea doar a unui tip de robot, robotul EV3. Dar, odată ce vă familiarizați cu acest tip de programare, puteți trece cu ușurință și la alt tip de robot.

Capitolul al III-lea introduce cititorul în problemele clasice ale roboticii prezentând posibile căi de rezolvare cu deschideri și către alte abordări.

Al IV-lea capitol oferă ca material didactic instrumente de evaluare și autoevaluare specifice activităților de învățare cu roboți.

Capitolul al V-lea se referă la mișcarea competițională existentă în România oferind modele de bună practică în învățarea roboticii.

Suportul de curs este în conformitate cu programa Informatică și TIC gimnaziu, Anexa nr. 2 la ordinul ministrului educației naționale nr. 3393 / 28.02.2017 care, referitor la robotică, cuprinde:

ROBOTICĂ EDUCAȚIONALĂ, CLASA A VII-A

Competența generală 1

Utilizarea responsabilă și eficientă a tehnologiei informației și comunicațiilor

Competența specifică 1.4

Utilizarea unui mediu de programare pentru implementarea algoritmilor

Exemple de învățare:

- explorarea elementelor de interfață ale unui mediu de programare în scopul identificării principalelor facilități ale acestuia
- aplicarea etapelor pentru dezvoltarea unei aplicații folosind un mediu de programare: editare cod sursă, compilare, executare, testare și depanare prin analogie cu mediile interactive care utilizează blocuri grafice
- utilizarea unui mediu virtual pentru programarea de roboți cu scop didactic, vizualizarea și utilizarea valorilor citite de senzori ai robotului virtual (de exemplu senzor ultrasonic pentru detectarea obstacolelor, senzor de culoare, senzor de presiune, microfon, senzor infraroșu, senzor giroscopic, busolă etc.)

ROBOTICĂ EDUCAȚIONALĂ, CLASA A VIII-A

Competența generală 1

1. Utilizarea responsabilă și eficientă a tehnologiei informației și comunicațiilor

Competența specifică 1.4

3.3. Implementarea algoritmilor într-un mediu de programare

Exemple de învățare:


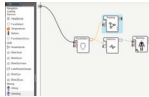


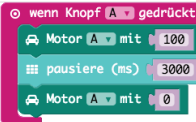


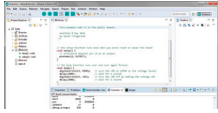





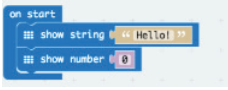


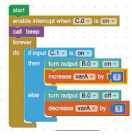
- elaborarea codului sursă pentru controlul robotului didactic virtual prin utilizarea și interpretarea datelor primite de la senzorii acestuia: evitarea obstacolelor, menținerea echilibrului, reacții specifice la detectarea luminii sau a identificării unui traseu marcat etc.



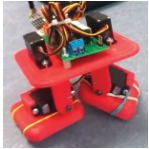







Tipuri de roboți

Un robot este format dintr-un dispozitiv programabil și o serie de dispozitive atașabile precum motoare, senzori și extensii folosite în construcția robotului. În robotica educațională, clasificarea roboților se face în funcție de dispozitivul programabil. Limbajul de programare poate fi sub formă de blocuri sau linie de cod. Este recomandat ca, pentru fiecare tip de robot, să fie folosit limbajul prevăzut în kit-ul de vânzare, dar pot fi utilizate cu succes și limbaje de programare disponibile pe platforme open source. Robotul poate fi programat să acționeze autonom sau comandat de la distanță.

ROBOT	DISPOZITIV PROGRAMABIL	LIMBAJ DE PROGRAMARE RECOMANDAT	LIMBAJE DE PROGRAMARE ALTERNATIVE
Nao (se citește now)		CHOREGRAPHE	Python, C++, MATLAB, Java, ROS, NEPO
			
Calliopi		PXT	JavaScript, NEPO
			
Arduino		Arduino IDE	ArduBlock, Minibloq, Java, C++, Arduino'C, Python, NEPO
			
Lego EV3		EV3	Scratch, EV3'C, NEPO
			
Micro:bit		JavaScript Blocks	JavaScript Blocks Editor (microbit-robot-jsb.hex), Python Editor (robot.py),
			
BOT'n ROLL		Picaxe blockly	Picaxe, C, Basic, NEPO
			



ROBOT	DISPOZITIV PROGRAMABIL	LIMBAJ DE PROGRAMARE RECOMANDAT	LIMBAJE DE PROGRAMARE ALTERNATIVE
BOB3 		ProgBob 	NEPO
NXT 		NXT 	NXC, Scratch, ATmega NEPO

OPEN ROBERTA – Prezentare generală

Open Roberta Lab este un mediu de programare online ce permite programarea roboților virtuali 2D și reali 3D. Prin mici proiecte veți înțelege cum interacționează un robot cu mediul înconjurător și cum veți putea să-i transmiteți acestuia comenzi astfel încât, împreună să duceți la bun sfârșit anumite acțiuni.

În România, cei mai populari roboți sunt cei de tip Lego, (EV3 și WEDO), Arduino (Arduino, Raspberry Pi) sau cei construiți cu piese tetrix utilizați, de regulă, în cluburile de robotică, în vederea pregătirii concursurilor de profil.

Platforma OPEN ROBERTA LAB, prin limbajul NEPO oferă posibilitatea de a programa atât roboți virtuali, folosind simulatorul integrat, cât și reali dacă aveți kit-urile achiziționate. Simulatorul integrat platformei permite programarea virtuală a unui model simplu de robot 2D pe un calculator cu o configurație simplă, conectat la Internet. Acest fapt face ca predarea roboticii educaționale să fie posibilă în toate școlile din România prin utilizarea resurselor actuale ale școlilor, fără costuri suplimentare, folosindu-se doar calculatoarele din dotarea laboratoarelor de informatică și conexiunea la Internet.

În suportul de curs am optat pentru programarea cu limbajul NEPO din OPEN ROBERTA LAB din următoarele motive:

- Programa școlară cuprinde programarea roboților ca parte din modul de programare;
- Programa școlară Informatică și TIC are ca obiectiv familiarizarea elevului cu diferite medii de programare, crearea competențelor de programare;
- OPEN ROBERTA LAB este o platformă gratuită;
- OPEN ROBERTA LAB nu necesită instalarea unor soft-uri suplimentare sau realizarea de conturi pentru elevi;
- Platforma poate fi folosită pentru roboți de tip EV3, NXT, MICRO:BIT, BOT n ROLL, NAO, BOB3, Calliope;
- Simulatorul OPEN ROBERTA LAB oferă robotul 2D deja echipat cu toți senzorii unui robot real pentru unitățile școlare care nu dispun de kit-uri sau nu au kit-uri suficiente pentru toți elevii;
- Programele din OPEN ROBERTA LAB pot fi transferate pe roboți reali de diferite tipuri, unitatea școlară putând opta, în funcție de costuri, pentru achiziționarea unui anumit tip de robot;



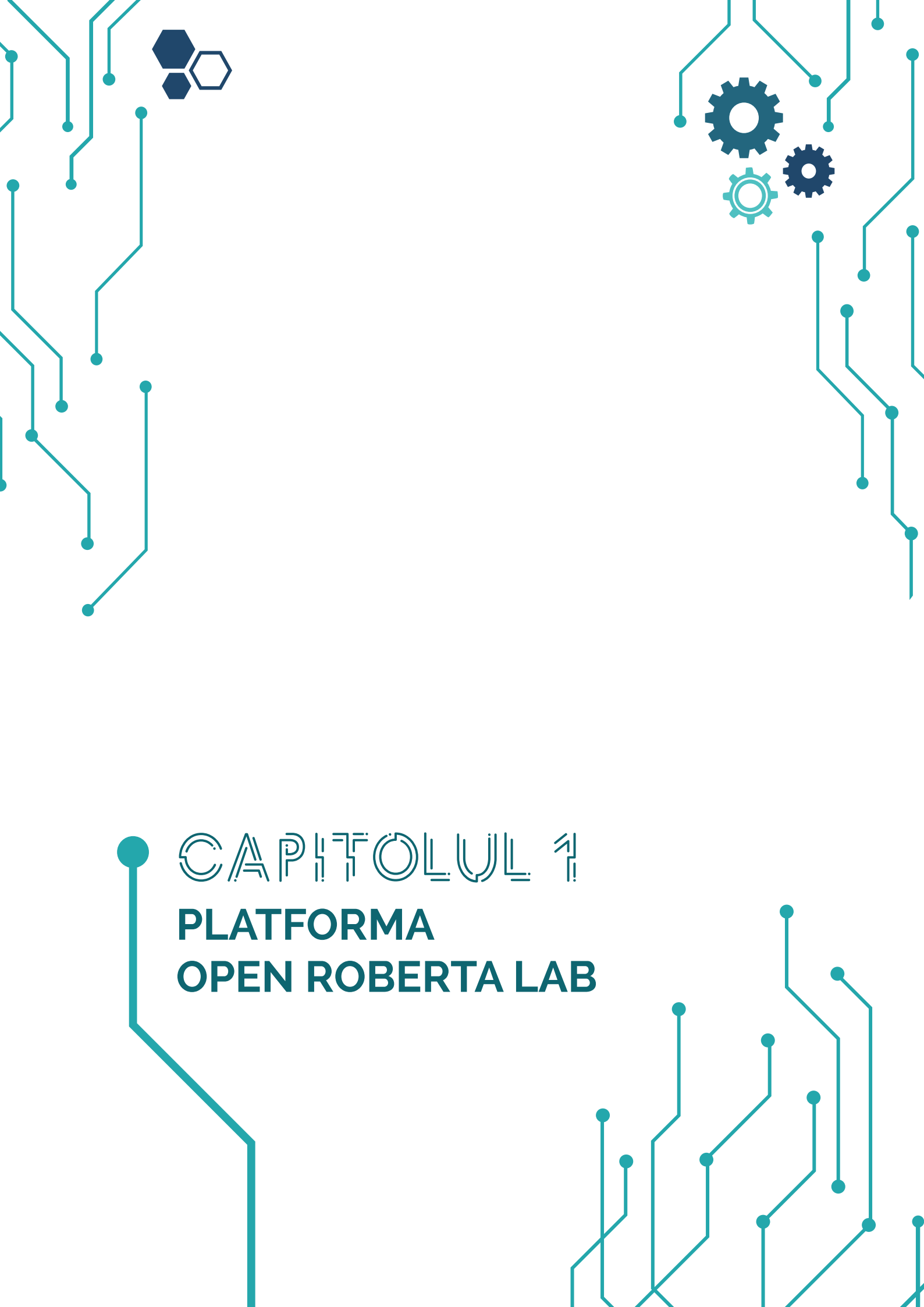
- Blocurile NEPO sunt intuitive, trecerea spre programarea în linie de cod, conform limbajelor de programare prevăzute în programa școlară, fiind mult mai ușoară;
- Blocurile NEPO sunt asemănătoare cu cele din Scratch putându-se fructifica experiența de programare din clasele anterioare;
- Față de mediul Scratch, OPEN ROBERTA LAB este dedicat programării roboților.

Cine a creat OPEN ROBERTA LAB?

Proiectată de Beate Jost, Reihard Budde, Thorsten Leimbach, Achim Kapusta, platforma Open Roberta a fost testată pentru prima dată în 2013, a apărut oficial în 2014, dar prima versiune stabilă, 2.0, a apărut în 11 octombrie 2016. Este rezultatul unui proiect educațional inițiat de Ministerul Educației și Cercetării din Germania în colaborare cu institutul de cercetare Fraunhofer IAIS și dezvoltat împreună cu Google. Poate fi utilizată gratuit de orice elev, profesor, sau utilizator fără restricții.

Open Roberta Lab se poate accesa folosind adresa <https://lab.open-roberta.org/>



The page is decorated with teal circuit lines and dots. A large teal dot is on the left, connected by a line to the text. Another large teal dot is at the bottom left, connected by a line to the text. The background is filled with various teal circuit patterns, including lines and dots, some of which are connected to the text.

CAPITOLUL 1

PLATFORMA OPEN ROBERTA LAB

Bună! Sunt Roberta și-ți voi arăta cum să înveți jucându-te cu roboți.



Utilizarea unui mediu de programare pentru implementarea algoritmilor

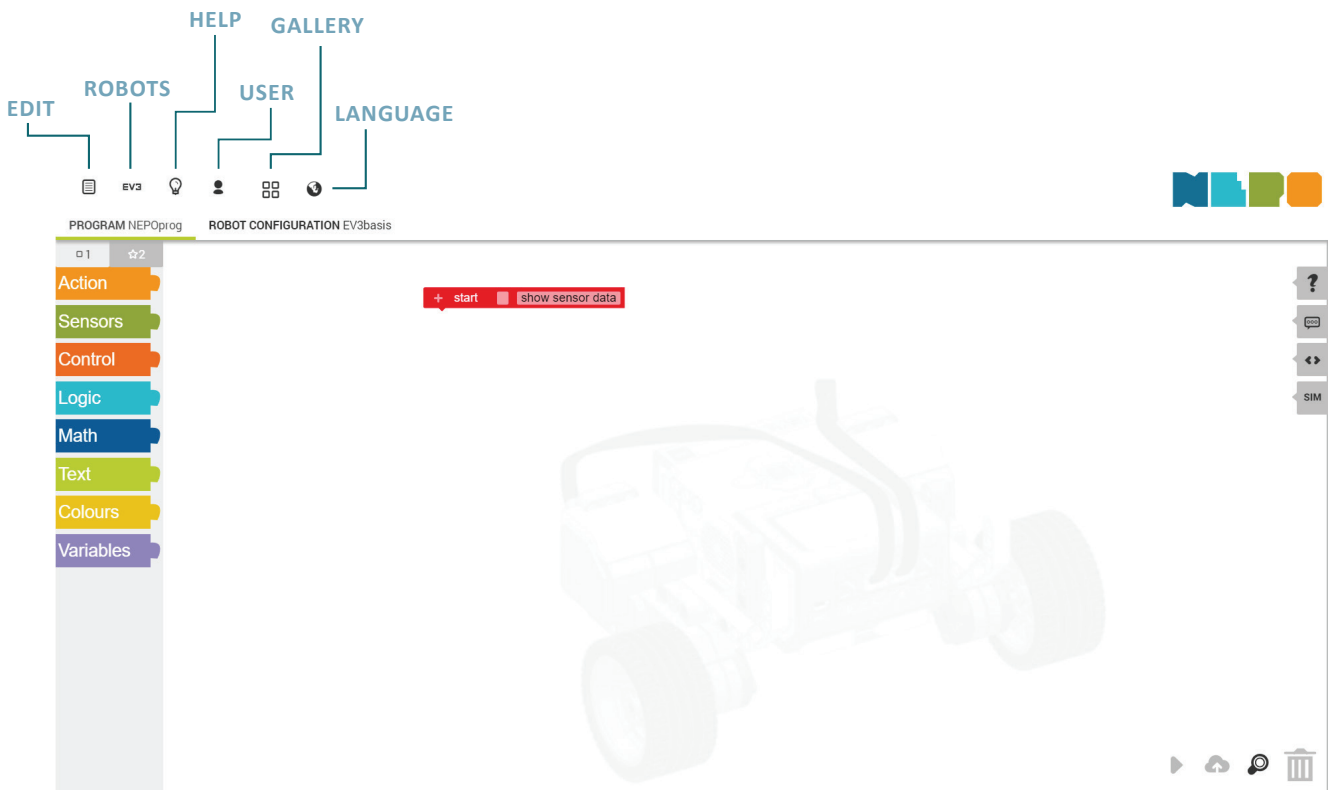
Explorarea elementelor de interfață ale unui mediu de programare în scopul identificării principalelor facilități ale acestuia

I. Platforma OPEN ROBERTA LAB

Meniurile platformei OPEN ROBERTA LAB

Fereastra principală conține în stânga sus 6 meniuri:

- Edit
- Robots
- Help
- User
- Gallery
- Language



EDIT

Acest element cuprinde un meniu în care veți găsi tot ce puteți face cu programul OPEN ROBERTA LAB: creare proiect, export/import proiect, exemple, vizualizare cod sursă etc. În plus, puteți alege blocurile NEPO pe care doriți să le programați, de exemplu, blocuri pentru începători sau blocuri pentru experți.





RUN ON ROBOT

Dacă faceți clic pe acest submeniu, robotul execută programul. Programul și configurația robotului vor fi trimise la serverele Open Roberta Lab unde este compilat și trimis robotului. Robotul rulează programul imediat ce este complet transferat. Dacă robotul real este conectat, va primi un mesaj direct.

RUN IN SIMULATION

Această comandă pornește programul în mediul de simulare. Programul va fi trimis la simulator și poate fi executat cu butonul de pornire al simulării. Configurația robotului este o configurație standard.

NEW ...

Creați un nou program. Dacă programul actual a fost modificat, va exista un avertisment că ar trebui să salvați modificările.

MY PROGRAMS ...

Afișează toate programele salvate. Pentru aceasta, trebuie să vă conectați la meniul user sau să creați un nou cont de utilizator cu new.

EXAMPLE PROGRAMS ...

Oferă exemple model pentru primele programe.

SHOW CODE

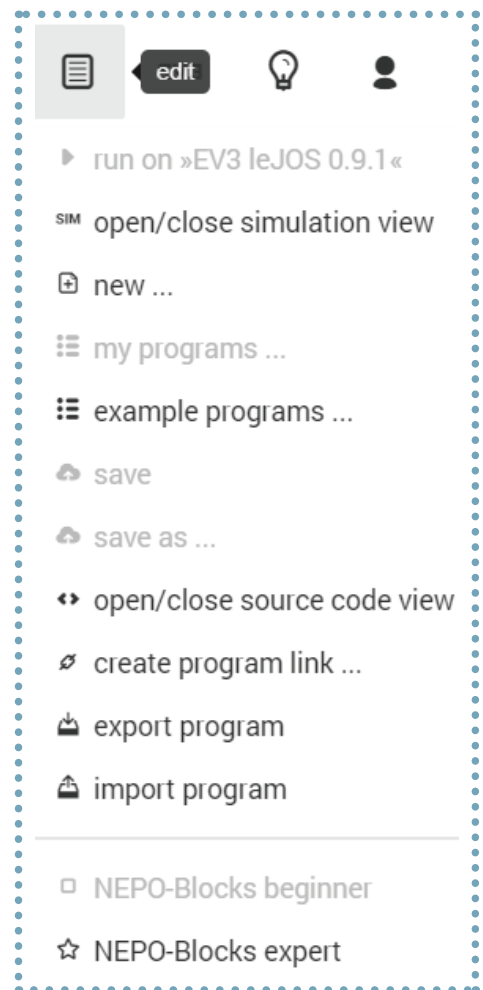
O fereastră de coduri va fi rulată pentru a afișa codul Java care a fost generat din programul cu blocuri.

CREATE PROGRAM LINK

Generează link-ul programului curent. Îl poți partaja sau include în alte pagini web.

EXPORT PROGRAM

Utilizarea comenzii export este folosită pentru a stoca programul curent pe un dispozitiv de stocare local. Codul XML generat va fi stocat în folderul dorit. Acest lucru este întotdeauna util dacă cloudul Open Roberta nu este disponibil.



SAVE

Dacă programul a primit deja un nume cu save as, se vor salva modificările programului. Această funcționalitate este disponibilă utilizând butonul din colțul din dreapta jos al ecranului.

SAVE AS ...

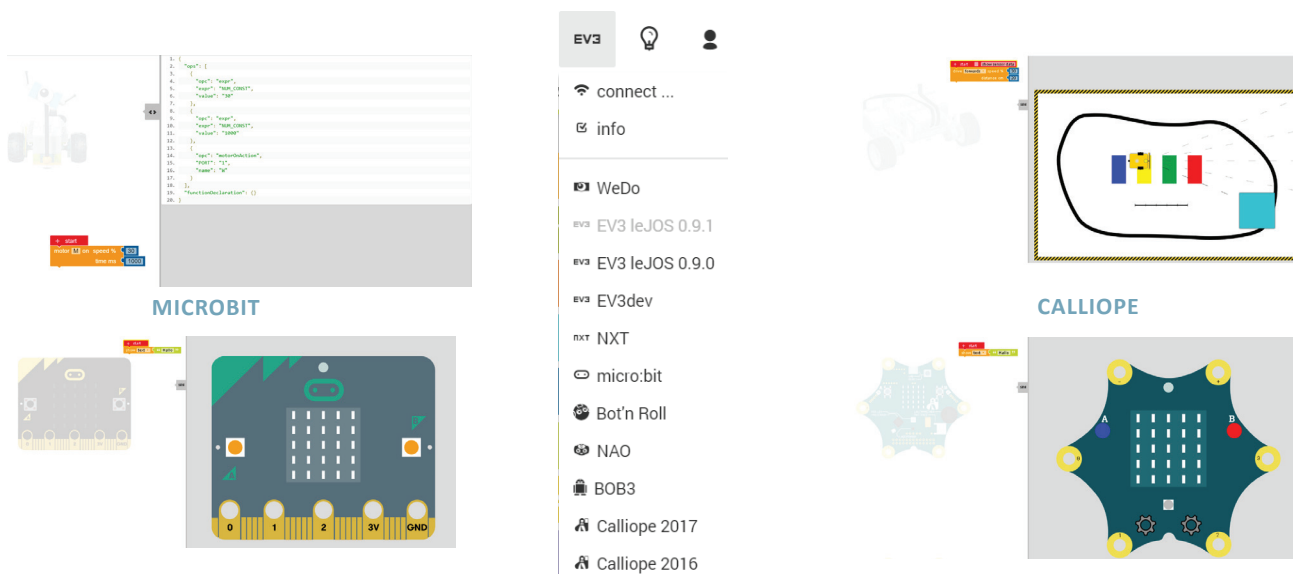
Programul e salvat cu un nume nou.



IMPORT PROGRAM

Comanda import este utilă pentru restaurarea unui program stocat pe un dispozitiv de stocare local.

Meniul Robots comută între simulatoarele diferitelor tipuri de roboți.



HELP MENU

Meniul Ajutor trimite spre Open Roberta Wiki. În plus, sunt disponibile următoarele submeniuri suplimentare:

GENERAL HELP

Aceasta este o legătură directă în wiki-ul Open Roberta, care oferă toată informația referitoare la proiect și programarea cu NEPO.

FAQ

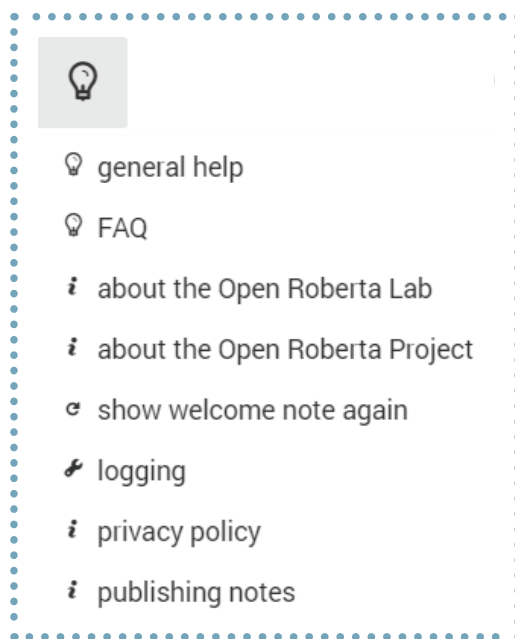
Întrebări frecvente și răspunsuri la acestea.

ABOUT

Aici puteți găsi câteva informații despre proiectul Open Roberta și despre Institutul Fraunhofer IAIS.

LOGGING

Afișarea tuturor acțiunilor efectuate în cadrul platformei Open Roberta Lab.



Dacă faceți clic pe acest submeniu, se va afișa din nou mesajul de întâmpinare. Aici sunt prezentate cele mai recente informații despre ultima versiune a platformei Open Roberta Lab.



USER

Utilizând acest element de meniu, vă puteți administra contul de utilizator în cadrul Open Roberta Lab. Pentru aceasta puteți utiliza următoarele submeniuri:

LOGIN

Realizează conectarea la un cont de utilizator existent. Dacă nu aveți un cont de utilizator, veți găsi un buton pentru a vă crea unul nou.

LOGOUT

Închide contul de utilizator, dar puteți încă programa și executa programe NEPO pe robot.

CHANGE

Modifică unele dintre datele contului de utilizator, cum ar fi numele de utilizator, numele afișat, adresa de e-mail sau parola.


DELETE USER

Șterge contul de utilizator și toate programele NEPO stocate.

STATE INFORMATION

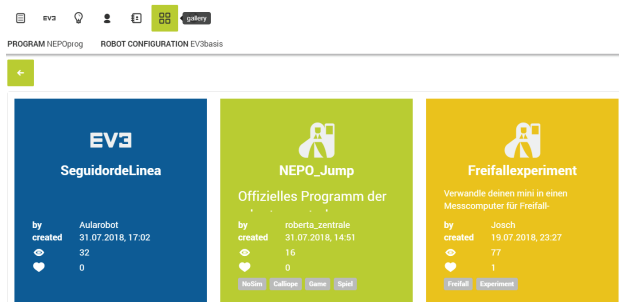
Afișează câteva informații de stare, cum ar fi numele de utilizator, numele programului, numele de configurare a robotului și modul de utilizare.

LANGUAGE MENU

Laboratorul Open Roberta este tradus în mai multe limbi. Puteți să alegeți limba în care va fi afișat programul apăsând butonul . Informațiile de ajutor sunt disponibile în prezent numai în limba engleză și în limba germană.

GALERIA OPEN ROBERTA LAB

Începând cu versiunea 2.3.0, Galeria este disponibilă permițând publicarea programelor într-o bibliotecă educațională. Utilizatorii conectați cu un cont pot face acest lucru făcând clic pe “editați” din bara de meniu, urmat de un clic pe punctul de meniu “programele mele”. La următoarea vizualizare a programelor salvate online, trebuie doar să faci clic pe pictograma galerie din partea dreaptă.



Pentru a folosi un program din Galerie, utilizatorii - conectați sau nu – pot da dublu clic pe programul ales și apoi pot vizualiza, schimba sau descărca conținutul.



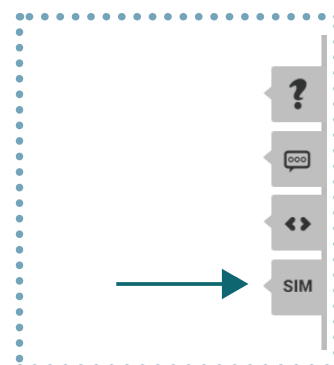
Simulatorul OPEN ROBERTA SIM

Mediul de simulare Open Roberta permite testarea programelor. Acest lucru este foarte util dacă nu există un robot real la îndemână.

Programele OPEN Roberta pot fi rulate pe roboți reali sau în mediul de simulare. În simulare este disponibil un model simplu de robot 2D.

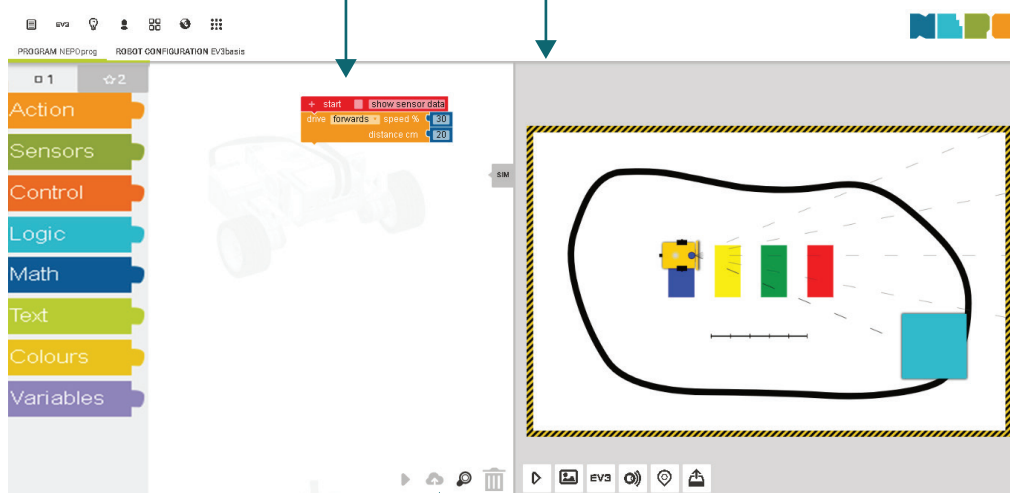
Mediul de simulare pentru robotul 2D va fi afișat făcând clic pe butonul SIM din dreapta mediului de programare. Același buton SIM poate fi utilizat pentru a închide mediul de simulare. În mediul de simulare există mai puține blocuri de programare executabile decât în mediul EV3. Acest lucru se datorează modelului robot simplu disponibil în prezent în mediul de simulare.

Când dați clic pe butonul de pornire a simulării pentru a rula un program în mediul de simulare, fereastra de afișare se modifică ca în figura de mai jos.



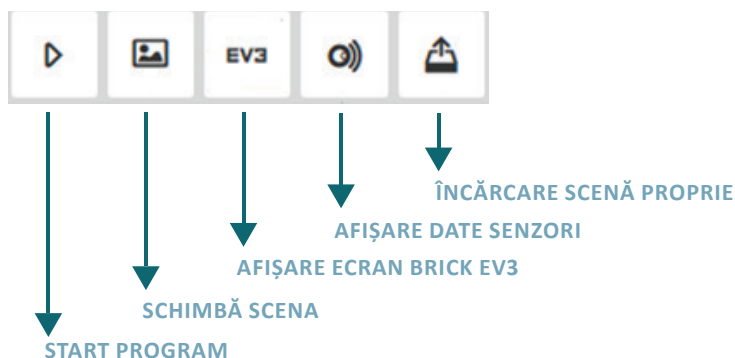
În panoul din stânga este afișată o parte a programului.

Panoul drept arată robotul 2D în mediul selectat în prezent.



În colțul din stânga jos există câteva butoane pentru a controla simularea.

BUTOANELE DE CONTROL ÎN OPEN ROBERTA SIM



În colțul din stânga jos al ferestrei de simulare există câteva butoane pentru a controla simularea robotului într-un anumit mediu definit de o imagine numită scenă. Butoanele disponibile în mediul de simulare sunt diferite în funcție de tipul de robot selectat dintre cei prevăzuți în mediul Open Roberta Lab.

Imaginea alăturată prezintă butoanele disponibile pentru modelul EV3.



▶ PORNIREA UNUI PROGRAM PE ROBOT

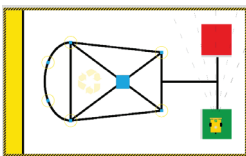
Pentru a verifica un program, faceți clic pe butonul de pornire din colțul din dreapta jos al ecranului ▶ sau EDIT - RUN.

Codul va fi compilat și trimis către robot pentru a fi executat. Veți primi un mesaj pe prima linie din fereastra Open Roberta care vă anunță când robotul va începe să execute programul.

SCHIMBĂ SCENA

Mediul de simulare poate fi schimbat. Există 6 medii standard prezentate în imaginile de mai jos, la care se pot adăuga medii personalizate adăugate de utilizator.

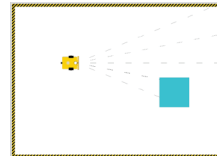
WORLD ROBOT OLYMPIAD SCENE



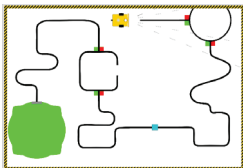
SIMPLE SCENE



DRAWING SCENE



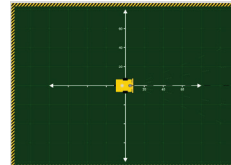
RESCUE SCENE





ROBERTA SCENE



MATH SCENE



Schimbarea scenei poate fi realizată când simulatorul este activat, fie folosind butonul  din stânga sus din ferestrei principale, fie folosind butonul  din bara ce apare în simulator în stânga jos.

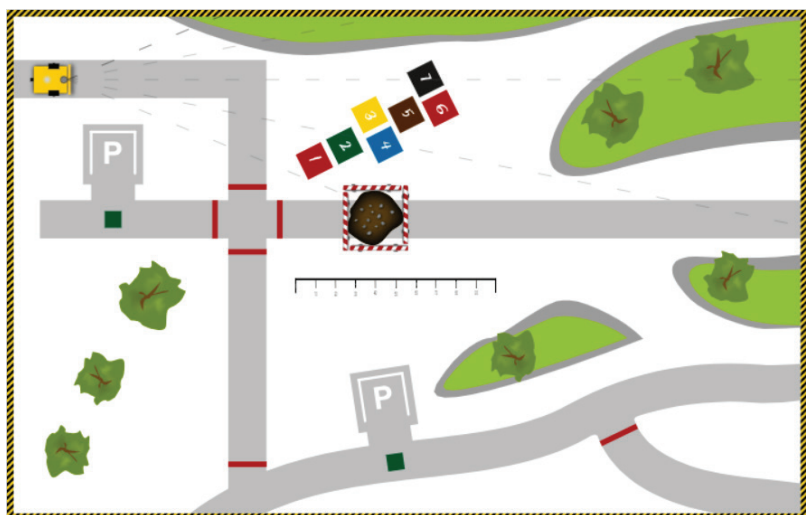
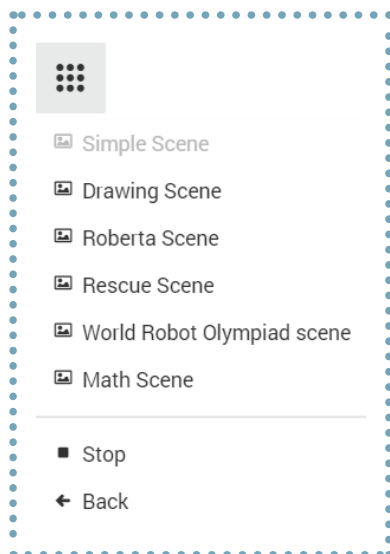


EV3



PROGRAM SeguidordeLinea Aularobot

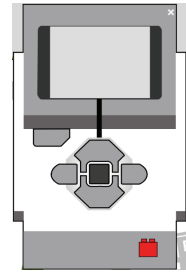
ROBOT CONFIGURATIC





EV3 BRICK-UL ROBOTULUI EV3

Făcând clic pe butonul **EV3** din mediul de simulare, se va afișa brick-ul robotului. Toate butoanele programabile ale brick-ului pot fi accesate cu mouse-ul. Lumina de stare a unui EV3 poate fi verde, portocalie sau roșie.



AFIȘARE DATE SENZORI

În funcție de mediul în simulare senzorii au valori diferite:

DRAWING SCENE

FPS	67
Time	3.68s
Robot X	153
Robot Y	275
Robot θ	95°
Motor left	0°
Motor right	0°
Touch Sensor	0
Light Sensor	100%
Ultra Sensor	72cm
Color Sensor	

WORLD ROBOT OLYMPIAD SCENE

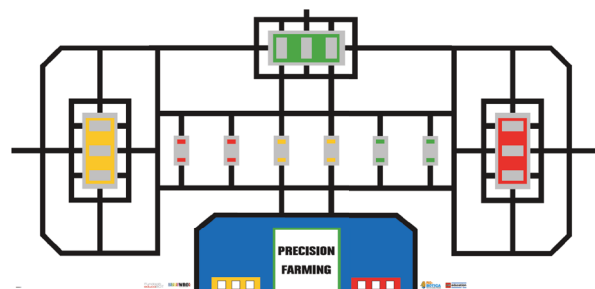
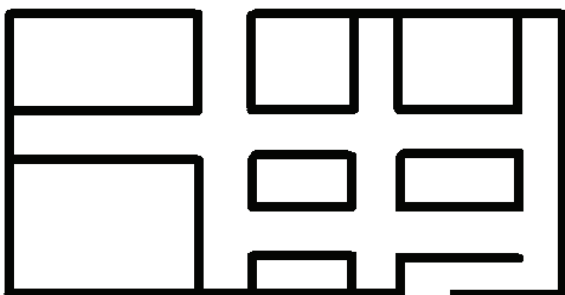
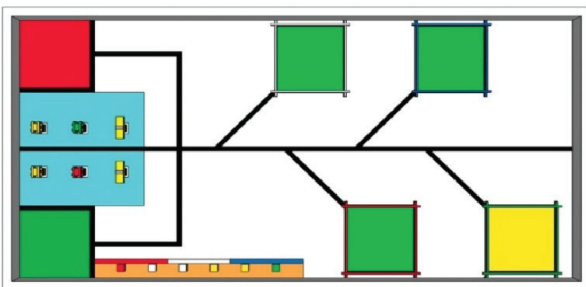
FPS	67
Time	0s
Robot X	800
Robot Y	440
Robot θ	270°
Motor left	0°
Motor right	0°
Touch Sensor	0
Light Sensor	26%
Ultra Sensor	96cm
Color Sensor	■



ÎNCĂRCARE SCENE PROPRII

Folosind butonul de încărcare, utilizatorul poate adăuga scene proprii, realizate cu diferite editoare grafice, în lista scenelor standard.

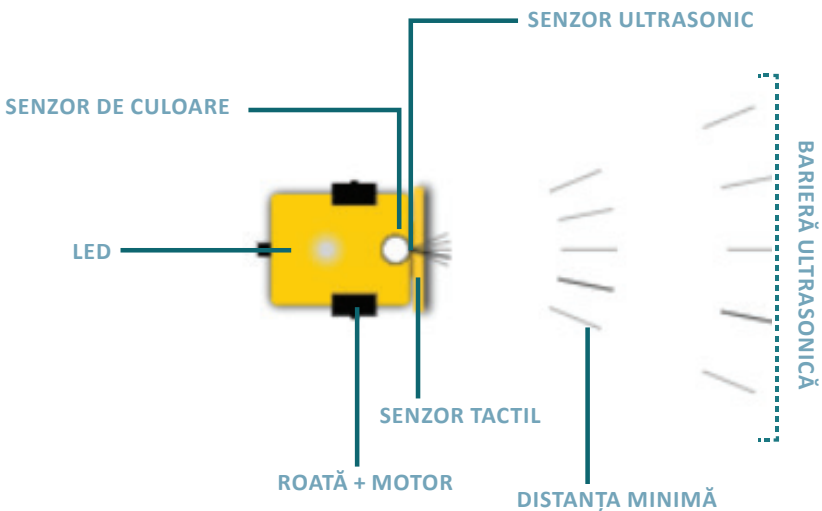
Exemple de scene personalizate:





ROBOTUL 2D ROBERTA EV3

Robotul 2D Roberta EV3 este format din următoarele componente ce pot fi utilizate în programele de simulare:



2 roți alimentate de motoare; 1 led care poate afișa lumină roșie, portocalie sau verde; 1 senzor de culoare pentru detectarea culorilor pe sol; 1 senzor ultrasonic în față; 1 senzor tactil integrat în bara de protecție

SETĂRI ROBOT 2D COMPATIBILE CU ROBOTUL EV3

EV3	
wheel diameter	5.6 cm
track width	18 cm
Sensor 1	touch sensor
Sensor 2	gyroscope
Sensor 3	colour sensor
Sensor 4	ultrasonic sensor
Motor A	big motor regulation <input type="checkbox"/> yes direction of rotation forwards side right
Motor B	
Motor C	
Motor D	big motor regulation <input type="checkbox"/> yes direction of rotation forwards side left

ROBOTUL 3D EV3



ROBOȚI EV3



BRICK-UL EV3 CU MOTOARE ȘI SENZORI

Setul EV3 include următorii senzori pentru robotul 3D:

Tactil - determină contactul cu suprafața

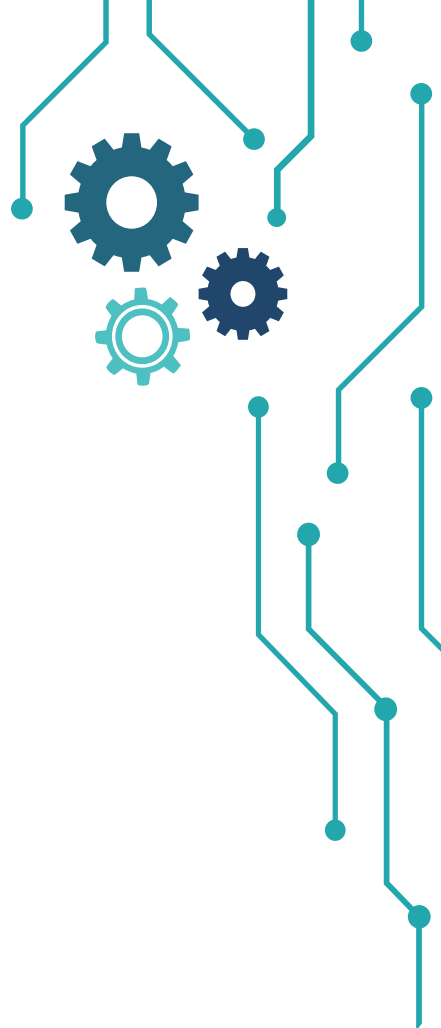
Ultrasonic - măsoară distanța față de suprafețele din apropiere

Culoare - determină culoarea și măsoară intensitatea culorii

Giroscopic - determină rotația robotului

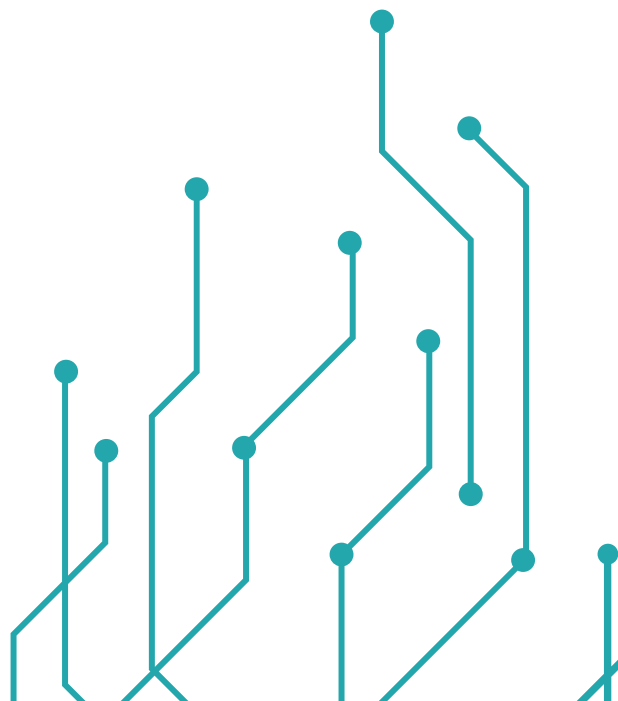
Infraroșu - determină semnalele telecomenzii IR





CAPITOLUL 2

LIMBAJUL NEPO



Bună! Sunt Roberta și-ți voi arăta cum să faci primul program.



Utilizarea unui mediu de programare pentru implementarea algoritmilor

Aplicarea etapelor pentru dezvoltarea unei aplicații folosind un mediu de programare: editare cod sursă, compilare, executare, testare și depanare prin analogie cu mediile interactive care utilizează blocuri grafice

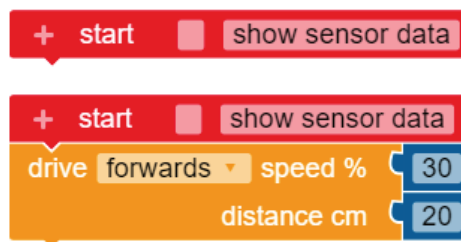
II. Limbajul NEPO

NEPO (New Easy Programming Online) este un limbaj de programare din cadrul platformei Open Roberta Lab care utilizează biblioteca Blockly la care au fost adăugate funcționalități suplimentare și îmbunătățiri, adaptate pentru Open Roberta Lab. Paradigma de programare a NEPO este inspirată de Scratch de la Massachusetts Institute of Technology.

Un bloc NEPO reprezintă și încorporează o anumită funcționalitate a robotului. Blocuri sunt organizate pe categorii, în funcție de acțiunile pe care le pot programa, de exemplu senzori. Blocurile sunt interconectate și vor fi executate de robot conform ordinii lor, începând cu blocul start. Acest principiu se numește programare secvențială.

În programarea cu NEPO, fiecare program începe cu blocul start. Acest bloc este adăugat întotdeauna în mod implicit în spațiul de lucru Open Roberta Lab. Primul bloc pe care îl veți executa este conectat cu conectorul de secvență al blocului start.

Conectorul de secvență este un triunghi în partea de jos a blocului și acest triunghi va deveni galben imediat ce un bloc adecvat este în imediata apropiere, după cum puteți vedea în imaginea alăturată.



BLOCURILE NEPO

Open Roberta Lab oferă două opțiuni atunci când selectează blocurile existente: Începător și Expert. Modul Începător este cel implicit. Această categorie include 8 blocuri care sunt cele mai importante pentru programarea unui robot. Modul Expert include în plus trei categorii de blocuri: "Liste", "Funcții" și "Mesaje".

CATEGORIE	DESCRIERE
ACTION	include blocuri pe care robotul poate să le execute direct
SENSORS	conține blocuri pentru toți senzorii standard ai sistemului EV3
CONTROL	include blocuri pentru controlul secvenței programului: dacă __ atunci __ dacă __ atunci __ altfel __ repețiți la infinit repețiți de n ori așteaptă așteaptă până când



CATEGORIE	DESCRIERE
LOGIC	cu blocurile logice pot fi create condiții cu ajutorul cărora se pot interconecta între ele stări, valori și evenimente
MATH	conține operatori matematici și blocuri de parametri
LIST	include blocuri pentru a crea o listă și pentru a căuta sau sorta elemente din listă
TEXT	include blocuri pentru a scrie texte pe afișajul robotului
COLOURS	conține blocuri standard de culoare pentru a compara intrările senzorilor
VARIABLES	conține blocuri cu ajutorul cărora pot fi definite variabile locale și globale
FUNCTIONS	conține blocuri cu ajutorul cărora pot fi definite funcții cu parametri de intrare și ieșire
MESSAGES	include blocuri pentru a trimite și a primi mesaje bluetooth
PROGRAM-START	fiecare program începe cu acest bloc; acest bloc este întotdeauna disponibil în spațiul de lucru

MENIUL CONTEXTUAL

Prin clic dreapta pe un bloc, poate fi deschis un meniu contextual. Utilizând meniul, pot fi efectuate mai multe acțiuni:

Duplicare (copiază blocul și parametrii de intrare conectați)

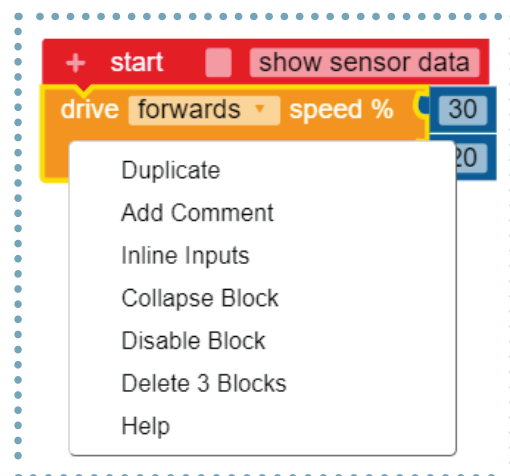
Adaugă comentariu (adaugă un comentariu la acest bloc)

Intrări inline (modifică aspectul blocului)

Restrângeți blocul (minimizează blocul)

Dezactivați blocul (dezactivează blocul astfel încât acesta să nu fie executat)

Ștergeți 3 Blocuri (șterge blocurile selectate și pe toate cele blocate)



CONECTORI NEPO DE INTRARE/IEȘIRE

Conectorul permite blocurilor NEPO să fie inserate în alte blocuri. În total, există șase tipuri de intrări și ieșiri în cadrul NEPO. Aceste tipuri pot avea următoarele culori:



TIP	CULOARE
VALOAREA LOGICĂ	albastru deschis
NUMĂR	albastru închis
STRING / TEXT	verde
CULOARE	galben
LISTĂ	purpuriu
CONEXIUNE	roz

PARAMETRII

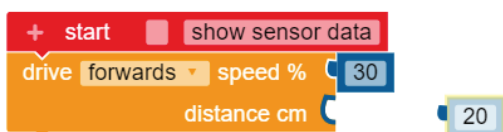
Valorile pot fi transferate de la un bloc la altul. Tipul valorii transferate poate fi identificat de conectorul unui bloc. În capitolul Categoriilor de blocuri puteți vedea diferitele valori pe care un bloc le poate avea. Blocurile pot fi conectate numai când culorile conectorului de intrare și ieșire se potrivesc. Un bloc poate, de asemenea, să treacă opțional o valoare (numai una) unui alt bloc. Aceste blocuri au conectorii de ieșire colorați.

O caracteristică specială a blocurilor de senzori în comparație cu blocurile de acțiune este faptul că multe blocuri de senzori returnează o valoare. Tipul de valoare returnat de un bloc de senzor corespunde culorii conectorului.



ACTIVITATE PRACTICĂ

În funcție de culoarea conectorilor, precizați tipurile de intrări/ieșiri pentru blocurile de mai jos.



Categoriile de blocuri

CATEGORIA ACȚIUNE (ACTION)

Categoria Acțiune conține blocurile care efectuează o acțiune asupra robotului LEGO MINDSTORMS EV3:

- Blocuri pentru controlul motoarelor
- Blocuri pentru a controla afișajul
- Blocuri pentru a controla lumina de stare
- Blocuri pentru a controla difuzorul



În modul expert, categoria Acțiune este împărțită în următoarele subcategorii:

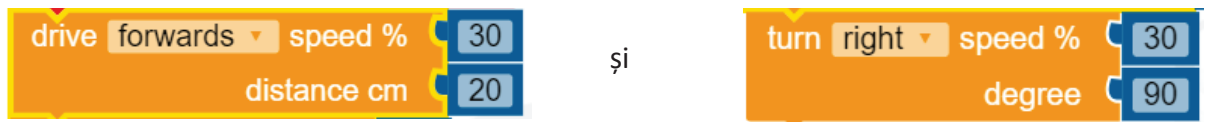
Mișcare • Conducere • Afișare • Sunete • Lumini



ACTIVITĂȚI PRACTICE

1.1. PĂTRAT

Realizați un program care va face ca robotul să traseze un pătrat. Folosiți blocurile pentru controlul motoarelor:

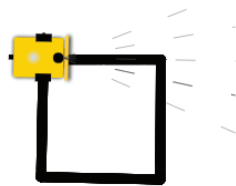
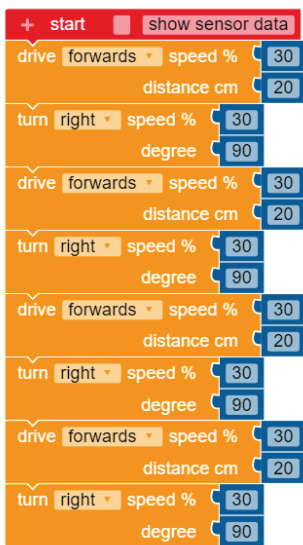


Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Drawing

1. CATEGORIA ACȚIUNE (ACTION)

SOLUȚIE

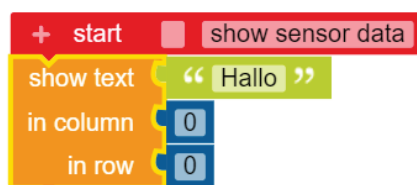


TEME DE DISCUȚII:

- Prin ce se aseamănă blocurile drive și turn?
- Prin ce se diferă blocurile drive și turn?
- Care este intervalul de valori pentru un unghi?

1. 2. ANA ARE MERE

Realizați un program care va afișa “Ana are mere” pe ecranul robotului. Folosiți blocurile pentru controlul afișajului și brick-ul robotului EV3.





Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Nu	Nu	În funcție de cerința problemei	Ecran EV3

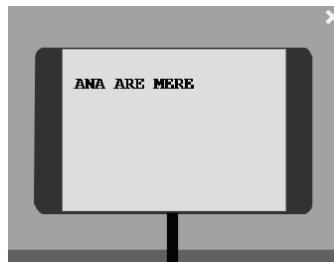
1. CATEGORIA ACȚIUNE (ACTION)

SOLUȚIE

```

+ start show sensor data
repeat until button enter pressed?
do
  show text "ANA ARE MERE"
  in column 1
  in row 1

```



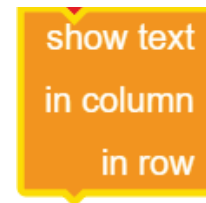
TEME DE DISCUȚII:
 Care sunt elementele blocului Show text?

1.3. CÂNTECEL

```

+ start show sensor data
play half note c

```



și brick-ul

Folosind blocurile Robotului EV3, realizați un program care va afișa și cânta gama.

Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Nu	Nu	În funcție de cerința din enunț	Ecran EV3

1. CATEGORIA ACȚIUNE (ACTION)

SOLUȚIE

```

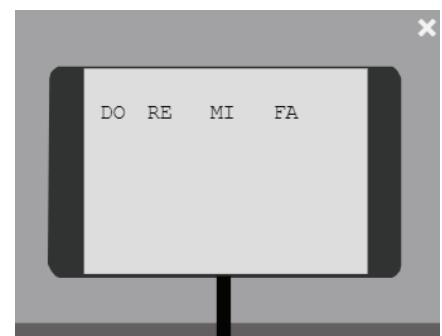
+ start show sensor data
show text "DO"
in column 1
in row 1
play half note c
show text "RE"
in column 4
in row 1
play half note d

```

```

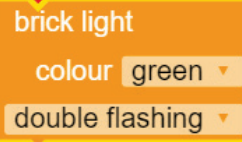
show text "MI"
in column 8
in row 1
play half note e
show text "FA"
in column 12
in row 1
play half note f

```





1.4. AVERTIZARE



Folosind blocul care controlează lumina de stare , realizați un program care să facă robotul să emită semnale luminoase atunci când se deplasează.

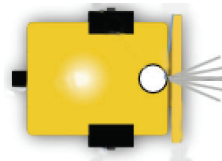
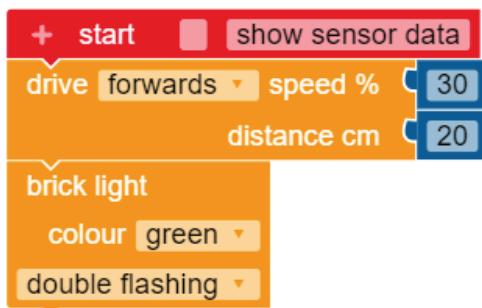
, realizați un program care să

Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	SIMULATOR
Roberta EV3	Da	Nu	Este recomandată scena Drawing

1. CATEGORIA ACȚIUNE (ACTION)

SOLUȚIE



1.5. CERC

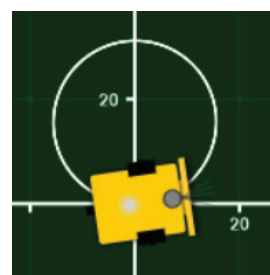
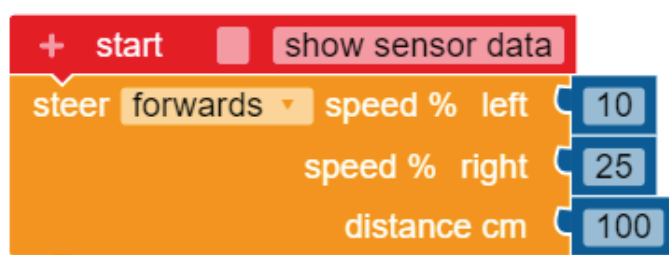
Folosind blocul care controlează roțile, realizați un program care să facă robotul să deseneze un cerc.

Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Numere intregi	Este recomandată scena Math

1. CATEGORIA ACȚIUNE (ACTION)

SOLUȚIE





APLICAȚII PROPUSE

- Realizați un program care va face ca robotul să emită semnale luminoase și acustice când se deplasează, ca o mașină de pompieri.
- Realizați un program care deplasează robotul în cerc pentru a desena o floare.
- Realizați un program care deplasează robotul pentru a desena un fluture.

CATEGORIA ACȚIUNE (ACTION)

Categoria Control include blocuri cu care secvența programului poate fi controlată.

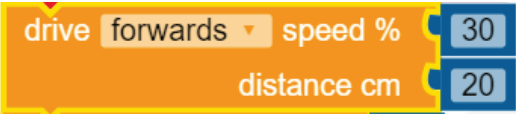
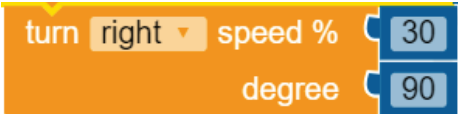

if do	dacă ___ atunci execută ___
if do else	dacă ___ atunci execută ___ altfel ___
repeat indefinitely	repetă la infinit
repeat n times	repetă de n ori
repeat while/until	repetă cât timp / până când
for each item in list	pentru fiecare element din listă
count with from to	numără de la ___ până la ___
break out/continue with next iteration of loop	oprește / continuă cu următoarea iterație de buclă
wait	așteaptă
wait until	așteaptă până când



ACTIVITĂȚI PRACTICE

2.1. PĂTRAT

Realizați un program care va face ca robotul să traseze un pătrat cu un număr minim de blocuri. Folosiți

blocurile  ,  și 

Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Drawing

2. CATEGORIA CONTROL

SOLUȚIE

```

+ start [show sensor data]
repeat 4 times
do
  drive forwards speed % 40 distance cm 40
  turn right speed % 10 degree 90
    
```

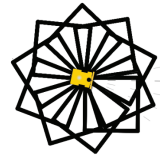




2.2. STEA

Realizați un program care va face ca robotul să traseze desenul din imaginea alăturată.

Folosiți blocurile:



Available blocks for the task:

- drive forwards speed % 30, distance cm 20
- turn right speed % 30, degree 90
- repeat 4 times do

Resurse necesare:

TIP ROBOT	MOTORE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Drawing

2. CATEGORIA CONTROL

SOLUȚIE

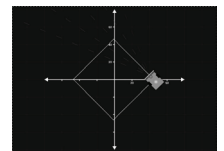
Solution code blocks:

- + start show sensor data
- repeat 10 times
 - do
 - repeat 4 times
 - do
 - drive forwards speed % 70, distance cm 50
 - turn right speed % 10, degree 90
 - turn right speed % 30, degree 35

2.3. ROMB

Realizați un program care va face robotul să traseze un romb ca în imaginea alăturată.

Folosiți blocurile:



Available blocks for the task:

- drive forwards speed % 30, distance cm 20
- turn right speed % 30, degree 90
- repeat 4 times do

Resurse necesare:

TIP ROBOT	MOTORE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Math





2. CATEGORIA CONTROL

SOLUȚIE

```

+ start  show sensor data
drive forwards speed % 30
  distance cm 45
turn left speed % 10
  degree 135
repeat 4 times
do
drive forwards speed % 30
  distance cm 65
turn left speed % 10
  degree 90

```

2.4 VIRAJE

Dacă un traseu are curbe, robotul trebuie să vireze la stânga sau la dreapta. Pentru simularea virajelor, în exemplul de mai jos se folosesc două blocuri corespunzătoare celor două roți în vederea setării parametrilor motoarelor asociate acestora. Motoarele standard în EV3 sunt notate cu B și C.

Folosiți blocurile:

```

motor port B on speed % 50 și wait ms 5000

```

Resurse necesare:

TIP ROBOT	MOTORE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Raza curbei este determinată de raportul vitezelor dintre motoarele B și C	Este recomandată Simple scene

2. CATEGORIA CONTROL

SOLUȚIE

```

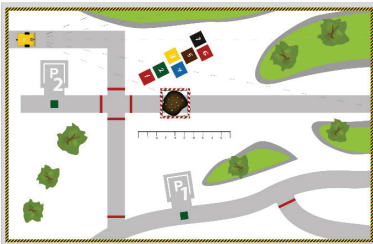
+ start  show sensor data
motor port B on speed % 50
motor port C on speed % 30
wait ms 5000


```




APLICAȚII PROPUSE

- Realizați un program care va face robotul să meargă pe traseul din imaginea de mai jos, scena Roberta.

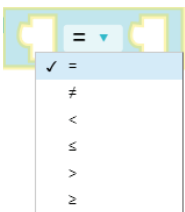


- » Prima destinație va fi obstacolul 
- » A doua destinație va fi P1.
- » Ultima destinație va fi P2.
- » Enumerați pașii pentru a ajunge la fiecare destinație și resursele de care aveți nevoie.

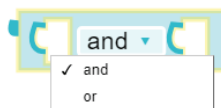
CATEGORIA LOGIC

Cu blocurile logice puteți crea condiții. Cu această condiție puteți interconecta stări, valori și evenimente între ele.

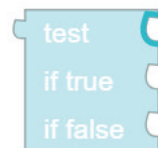
BLOCUL COMPARĂ



BLOCUL ȘI/SAU



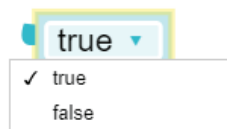
BLOCUL TEST



BLOCUL NEGAȚIE



BLOCUL ADEVĂRAT/FALS



BLOCUL NULL



ACTIVITĂȚI PRACTICE

3.1. COMPARĂ

Realizați un program care va face robotul să înainteze până când se apasă tasta Enter.

Folosiți blocurile:



Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție enunț.	Este recomandată scena Drawing



3. CATEGORIA LOGIC

SOLUȚIE

```

+ start [ ] show sensor data
drive forwards speed % [ 30 ]
+ wait until [ get pressed button enter ] = [ true ]
    
```



APLICAȚII PROPUSE

- Realizați un program care va face robotul să se rotească în cerc până când se apasă tasta ESC.
- Realizați un program care va face robotul să se deplaseze 5 secunde. Folosiți blocul:

```

get value ms timer [ 1 ]
    
```

- Realizați un program care va face robotul să se deplaseze 50 cm. Folosiți blocul:

```

get distance cm encoder [ B ]
    
```

CATEGORIA MATEMATICĂ (MATH)

Folosind blocurile matematice, utilizatorul poate genera expresii, calcula valori, valida date și seta o valoare implicită. În limbajul NEPO sunt disponibile următoarele blocuri matematice în modul Începător:

PARAMETRU

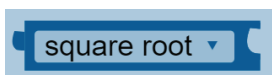


DE CALCULAT



În modul Expert din limbajul NEPO sunt disponibile următoarele blocuri:

FUNCȚIE MATEMATICĂ



SCHIMBAȚI CU X



FUNCȚIA TRIGONOMETRICĂ



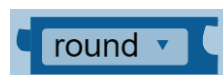
RESTUL



CONSTANTE



ROTUNJIRE



NUMĂRUL PROPRIETĂȚII



LISTA DE EVALUARE





CONSTRÂNGETI

constrain low 1 high 100

FRAȚIUNE ALEATORIE

random fraction

ALEATORIU ÎNTREG

random integer from 1 to 100



ACTIVITĂȚI PRACTICE

4.1. NUMERE ALEATOARE

Generați numere aleatoare de la 1 la 100.

Folosiți blocurile:

show text in column in row random integer from 1 to 100 wait ms 2500

Resurse necesare:

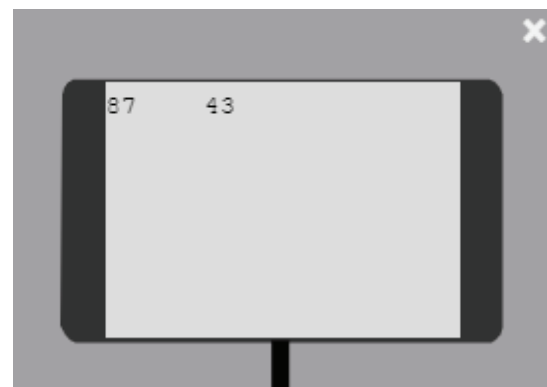
TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție de rezultatul dorit.	Ecran EV3

4. CATEGORIA MATH

SOLUȚIE

```

+ start show sensor data
show text random integer from 1 to 100
in column 0
in row 0
show text random integer from 1 to 100
in column 5
in row 0
wait ms 2500
    
```



4.2. BISECTOARE

Realizați un program care va face ca robotul să traseze o linie de la (0,0) la poziția (40, 40).

Folosiți blocurile:

drive forwards speed % 30 distance cm 90 turn left speed % 30 degree 20

40 ^ 2






Resurse necesare:

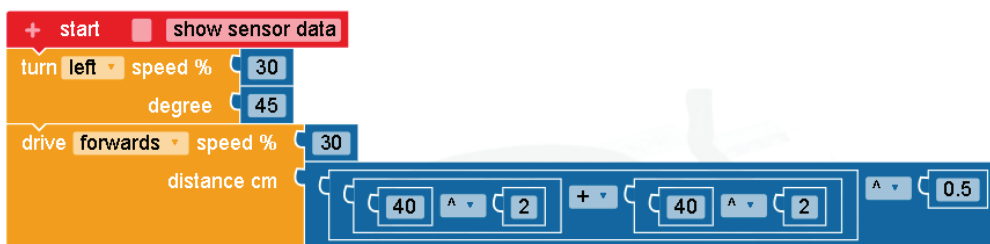
TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Math

Robotul trebuie să parcurgă distanța exactă de 56,57 cm cu un unghi de 45 °. Cum ajungem la distanța de 56,57 cm? Pentru a afla distanța pe care trebuie să o parcurgă robotul, folosim teorema lui Pitagora $c^2 = a^2 + b^2$ pentru $a = 40$ și $b = 40$. Deoarece nu există nici un bloc matematic pentru rădăcină pătrată, vom folosi funcția exponențială cu parametrul 0,5: $c = (a^2 + b^2)^{0,5}$.

Folosim blocul calcul  pentru a aduna, ridica la putere și pentru a extrage rădăcina pătrată: $((40^2) + (40^2)^{0,5}) = 56,57$.

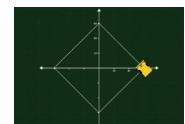
4. CATEGORIA MATH

SOLUȚIE

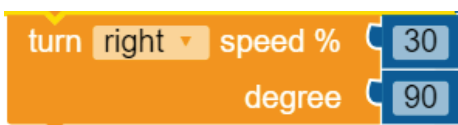
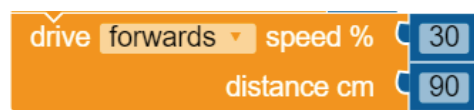


4.3. ROMB

Realizați un program care va face robotul să traseze un romb cu diagonalele egale cu 120 cm, ca în imaginea alăturată.



Folosiți blocurile:



Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Nu	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Math

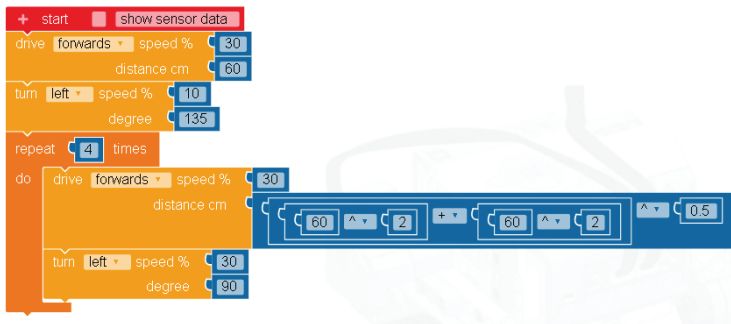




4. CATEGORIA MATH

SOLUȚIE

Folosind calculul matematic vom obține o soluție îmbunătățită față de soluția 2.3 obținută prin încercări.



APLICAȚII PROPUSE

- Generați 15 numere aleatoare de la 301 la 400 și afișați-le câte 5 pe linie. Explicați rolul și valorile parametrilor din blocul random. Enumerați blocurile pe care le folosiți și rolul acestora. Prezentați rezultatul în fața colegilor.
- Desenați un brad. Explicați pașii pe care robotul îi are de urmat. Enumerați blocurile pe care le folosiți și rolul acestora. Prezentați rezultatul în fața colegilor.
- Care dintre soluțiile următoare credeți că ar fi mai simplă pentru realizarea programului?
 1. Desenăm 3 triunghiuri de laturi diferite.
 2. Desenăm conturul bradului după ce am calculat laturile și unghiurile.

CATEGORIA TEXT

Blocurile de text efectuează operații simple pe șiruri de text.

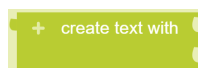
ÎNCEPĂTOR



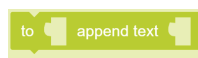
TEXT



EXPERT



CREAȚI TEXT (BLOC EXPERT)



ADĂUGAȚI TEXT (BLOC EXPERT)

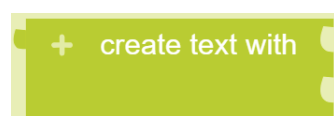
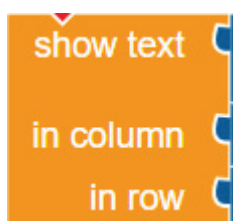


ACTIVITĂȚI PRACTICE

TEXT COMPUS

Realizați un program în care un mesaj e compus din text, număr și culoare.

Folosiți blocurile:



Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Nu	Nu	Parametrii trebuie să fie ajustați în funcție de enunț	Brick EV3

Pot fi realizate texte combinând orice tip de dată din lista de mai jos:

- » Valoarea logică (albastru deschis)
- » Număr (albastru închis)
- » String / Text (verde)
- » Culoare (galben)
- » Listă (purpuriu)



APLICAȚII PROPUSE

- Afișați pe ecranul EV3 tabelul de adevăr pentru operatorul AND:

```
True And True = True
True And False = False
False And True = False
False And False = False
```

- Afișați pe ecranul EV3 tabelul de adevăr pentru operatorul OR:

```
True Or True = True
True Or False = True
False Or True = True
False Or False = False
```

CATEGORIA VARIABLE (VARIABLES)

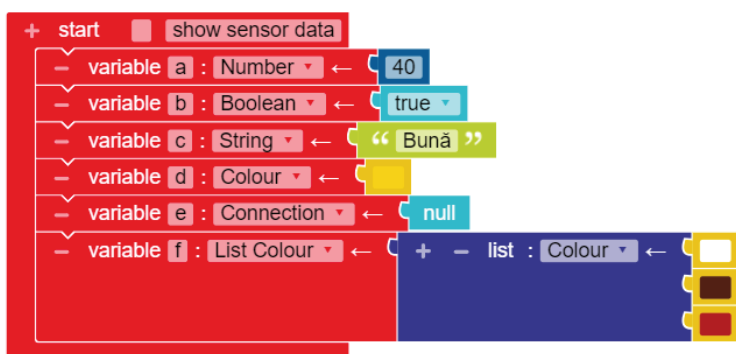
Blocurile Variabile sunt folosite pentru a crea variabile utilizate în programele NEPO ca parametrii în blocurile condiționale sau repetitive. Pentru a crea o variabilă, faceți clic pe semnul + din blocul start. Fiecare variabilă trebuie creată înainte de a fi utilizată. Fiecare variabilă trebuie să aibă un nume și un tip ce poate fi număr, boolean, șir, culoare, conexiune, listă. În numele variabilei nu sunt permise caracterele speciale și semnul spațiu. În limbajul NEPO sunt disponibile pentru categoria variabile următoarele blocuri:



SETEAZĂ VARIABILĂ



OBȚINE VARIABILĂ



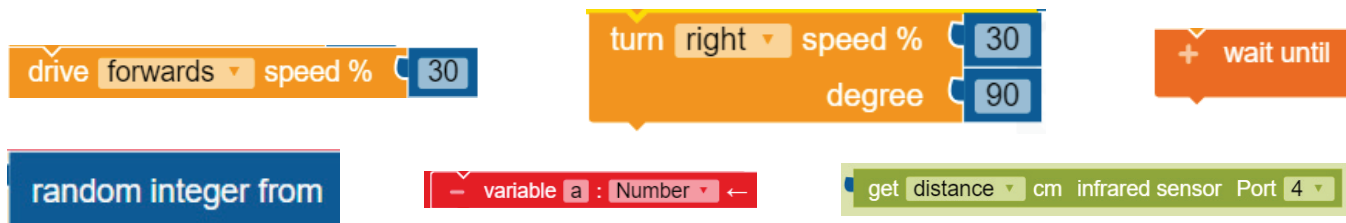


ACTIVITĂȚI PRACTICE

6.1. OBSTACOL

Realizați un program care va face ca robotul să se întoarcă cu 180 grade la X cm distanța față de locul de plecare, unde X este un număr cuprins între 25 și 70.

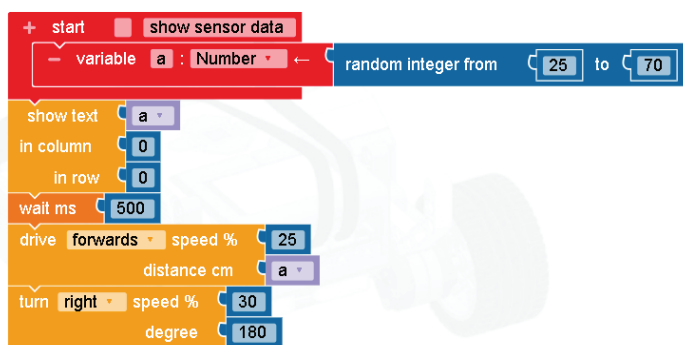
Folosiți blocurile:



Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Da	Ultrasonic	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Math

SOLUȚIE



APLICAȚII PROPUSE

- Realizați un program care va afișa pe ecranul Ev3 tabla înmulțirii cu X, unde X este un număr cuprins între 1 și 9.
- Realizați un program care va afișa pe ecranul Ev3 aria pătratului cu latura X, unde X este un număr cuprins între 20 și 100.
- Realizați un program care va afișa X cercuri de rază R, unde X este un număr cuprins între 1 și 9 și R un număr cuprins între 1 și 100.

CATEGORIA LISTE (LIST)

O listă este o secvență de date de același tip într-o succesiune fixă. Cu blocul liste din NEPO se pot realiza variabile de tip listă ce pot conține numere, texte, boolean, culori și conexiune, imagini.



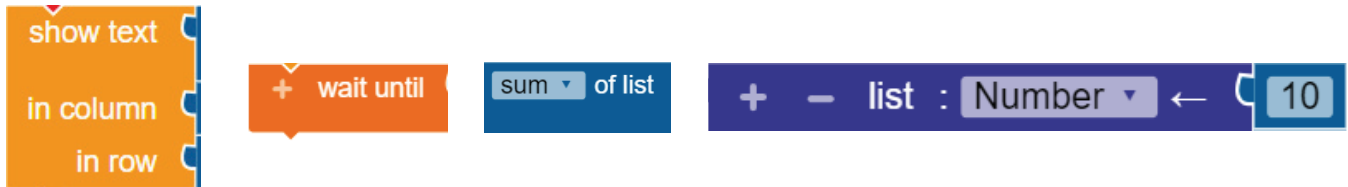


ACTIVITĂȚI PRACTICE

7.1. SUMA

Realizați un program care va calcula suma numerelor dintr-o listă dată.

Folosiți blocurile:

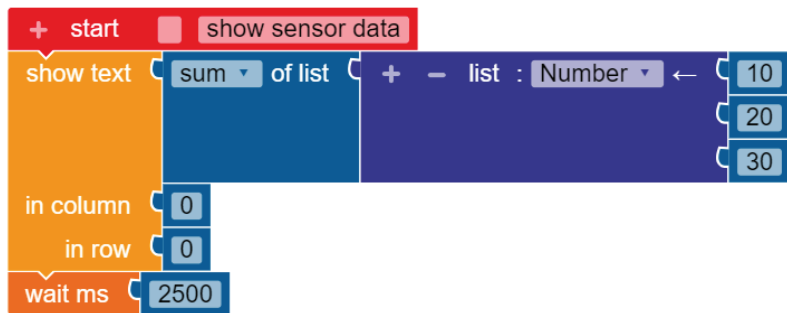


Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Nu	Nu	Parametrii trebuie să fie numere întregi.	Ecran EV3

7. CATEGORIA LISTE

SOLUȚIE

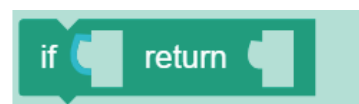
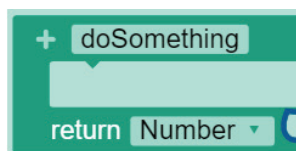
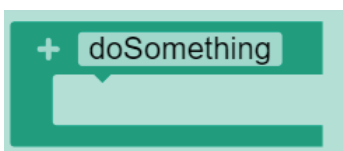


APLICAȚII PROPUSE

- Realizați un program care să calculeze suma numerelor dintr-o listă de 3 variabile a, b, c.
- Afișați culorile dintr-o listă de culori.
- Verificați dacă culoarea RED este prima într-o listă de culori.
- Creați o variabilă de tip listă și afișați pe ecranul EV3 valorile din lista numerică.

CATEGORIA FUNCȚII (FUNCTIONS)

Cu blocurile Funcții puteți crea programe complexe mai ușor de citit. NEPO oferă câteva blocuri de funcții:





În exemplul de mai jos funcția suma calculează suma a două variabile a și b.

CATEGORIA MESAJE (MESSAGES)

Doi roboți EV3 pot schimba mesaje. Este necesar ca Bluetooth să fie activat pe roboții EV3 participanți. Unul dintre roboții implicați stabilește conexiunea cu celălalt robot (bloc expert: se conectează la numele robotului).

Celălalt robot așteaptă un apel de conectare (bloc expert: așteptați conexiunea). Aceasta funcționează similar cu apelurile telefonice: un apelant formează numărul unui partener de comunicare. Când este stabilită conexiunea, roboții pot schimba mesaje în ambele direcții.

Bună! Sunt Roberta și-ți voi arăta cum să faci primul program folosind senzori.



Utilizarea unui mediu de programare pentru implementarea algoritmilor

Utilizarea unui mediu virtual pentru programarea de roboți cu scop didactic, vizualizarea și utilizarea valorilor citite de senzori ai robotului virtual (de exemplu senzor ultrasonic pentru detectarea obstacolelor, senzor de culoare, senzor de presiune, microfon, senzor infraroșu, senzor giroscopic, busolă etc.)

CATEGORIA SENZORI (SENSORS)

OPEN ROBERTA LAB conține blocuri pentru următorii senzori:

- » senzor de atingere
- » senzor de culoare/lumina
- » senzor ultrasonic
- » senzor de infraroșu
- » senzor de rotație (codificator motor)
- » giroscopic
- » butoane
- » timer






CATEGORIA CULORI (COLORS)

Categoria culori oferă blocurile de culoare necesare senzorului de culoare.



Senzorul de culoare/lumină

Cu blocul  se poate transmite unui alt bloc care este culoarea senzorului. În plus, acest bloc oferă în meniul drop-down setările lumină (light), RGB și lumina ambientală (ambient light).



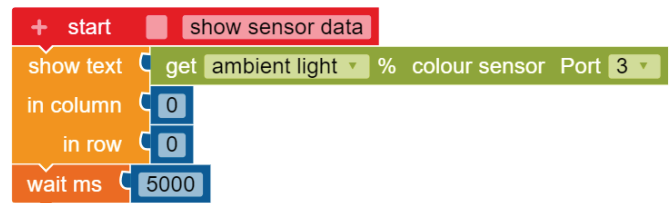
Aceste trei setări suplimentare transmit toate aceleași date de tip numeric. Valorile numerice sunt între 0 (negru) și 100 (alb). Cu aceste setări diferite, acest bloc poate fi configurat în funcție de cerințele specifice.

În modul color, senzorul emite lumină și detectează culoarea de bază de sub senzor. Recunoaște 7 culori (negru - BLACK, maro - BROWN, albastru - BLUE, verde - GREEN, galben - YELLOW, roșu - RED, alb - WHITE) și fără culoare - GRAY.

În modul lumină, senzorul emite lumină prin LED-ul său roșu și măsoară intensitatea luminii reflectate pe o scară de la 0 la 100, (0 = foarte întunecată și 100 = foarte luminoasă).

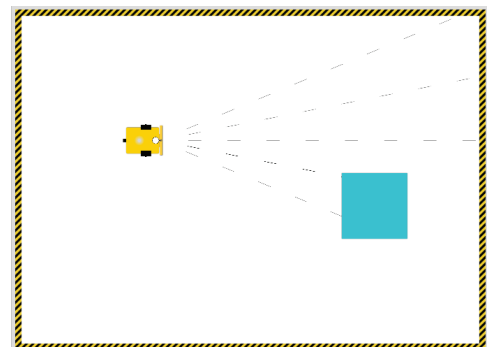
În modul lumină ambientală se utilizează aceeași scară (0-100) ca în modul lumină. În imaginea alăturată se măsoară lumina ambientală care este percepută de senzor.

În modul RGB, senzorul de lumină emite lumină roșie, verde și albastră și măsoară lumina reflectată corespunzătoare componentelor roșu, verde și albastru pe o scară de la 0 la 100 pentru fiecare culoare. Se întoarce o listă cu trei numere.



Senzorul tactil/de atingere

Cu blocul senzor tactil/de atingere puteți transmite unui alt bloc valori dacă senzorul tactil este apăsat sau nu. Acest bloc returnează valorile logice true dacă e în modul apăsat sau false dacă nu e apăsat. Acest bloc poate fi utilizat numai împreună cu un alt bloc care necesită o valoare logică ca parametru de intrare, de exemplu împreună cu blocul dacă da...atunci.



În mediul OPEN Roberta LAB senzorul tactil este apăsat dacă întâlnește un obstacol. În fiecare scenă, marginea este un obstacol fix, pătratul albastru din Simple și Drawing Scene este un obstacol mobil ce poate fi deplasat prin drag and drop.





Senzorul ultrasonic

Cu blocul **obține distanța/prezența** se transmite unui alt bloc distanța măsurată de senzorul ultrasonic. Distanța este transmisă ca un număr, în cm.

În plus, acest bloc poate fi setat la “prezență” din meniul derulant. Această setare poate verifica dacă un alt senzor ultrasonic este activ. Această setare este specifică valorii logice true dacă un alt senzor ultrasonic este prezent sau false dacă niciun alt senzor ultrasunet nu este prezent.

Dacă valoarea nu a fost trimisă de propriul senzor ultrasonic, poate duce la măsurători eronate. Dacă a fost selectat “distanța”, un număr care indică distanța în cm dintre senzorul ultrasonic și obstacol, se transmite unui alt bloc.



ACTIVITĂȚI PRACTICE

11.1. Realizați un program care face ca robotul să emită un semnal sonor la întâlnirea unui obstacol.

SOLUȚIE

```

+ start
show sensor data
repeat until touch sensor Port 1 pressed?
do
  drive forwards speed % 30
  distance cm 20
play whole note c'

```

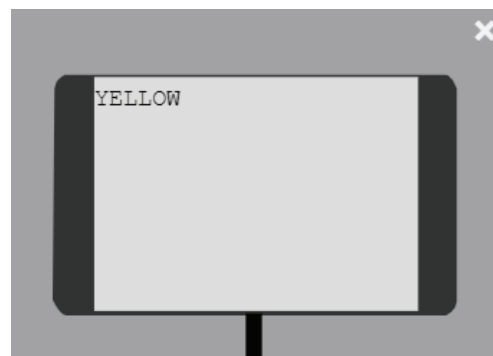
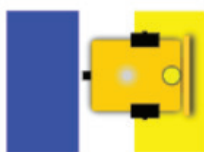
11.2. Afășează pe ecranul robotului culorile pe care le recunoaște senzorul de culoare.

SOLUȚIE

```

+ start
show sensor data
show text get colour colour sensor Port 3
in column 0
in row 0
wait ms 1500
clear display

```



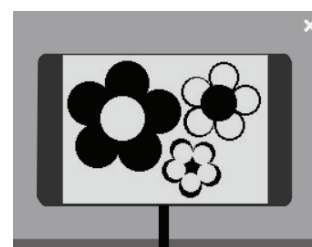
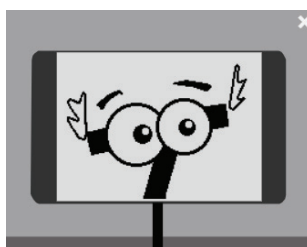
11.3. Realizați un program care afășează imaginea flowers până când senzorul tactil este apăsat. După ce s-a oprit robotul, pe ecran să apară imaginea eyes open.

SOLUȚIE

```

+ start
show sensor data
repeat until touch sensor Port 1 pressed?
do
  show picture flowers
show picture eyes open
wait ms 1500

```





11.4. Realizați un program care, utilizând senzorul de culoare, să deplaseze robotul până la o linie verde.

SOLUȚIE

```
+ start [ ] show sensor data [ ]
drive forwards [ ] speed % [ 30 ]
+ wait until [ ] [ get colour [ ] colour sensor Port 3 [ ] = [ ] [ ]
```



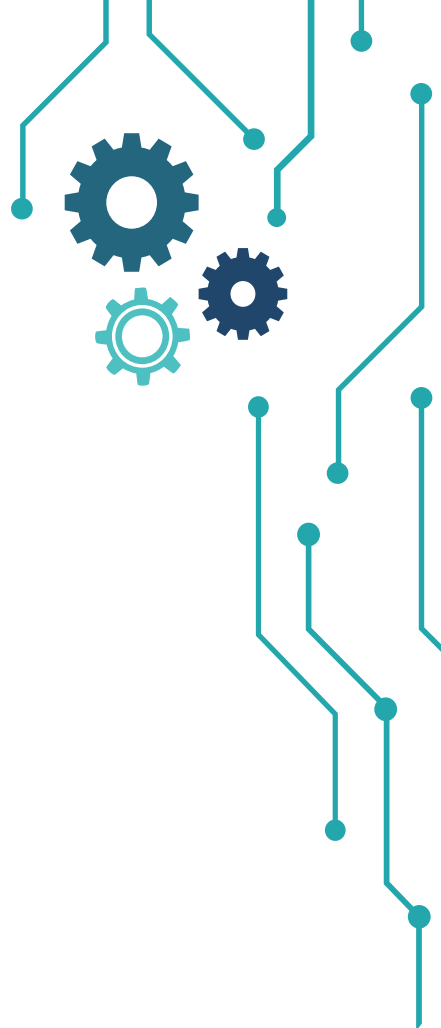
APLICAȚII PROPUSE

- La întâlnirea unui obstacol robotul să emită un semnal sonor.
- Robotul să se oprească când senzorul cu ultrasunete este la 20 cm distanță de un obstacol.

TEME DE DISCUȚII:

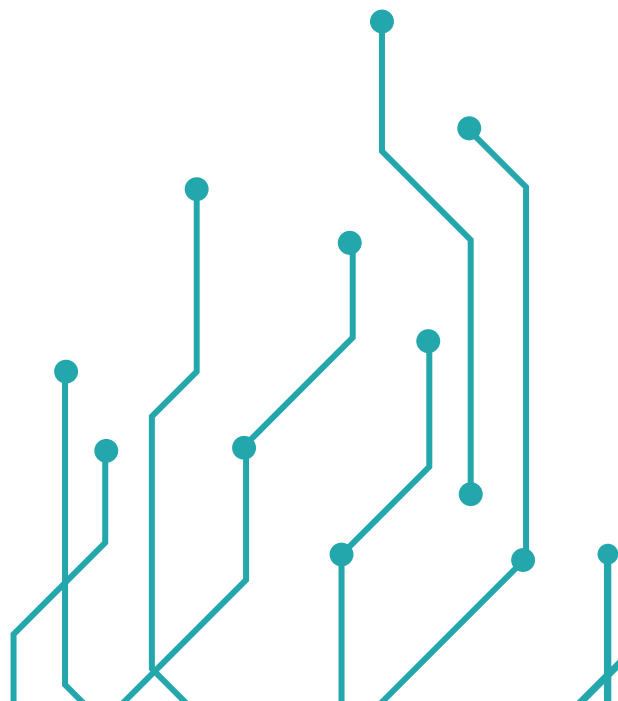
- Puteți căuta mai mult de o culoare cu senzorul de culori?
- Puteți identifica obiecte folosind senzorul ultrasonic?





CAPITOLUL 3

LECȚII



Bună! Sunt Roberta și-ți voi arăta cum să faci primul program pentru un concurs



TEME DE DISCUȚII:

Cum se deplasează oamenii pe o cărare. Preferă să meargă pe mijlocul drumului? De ce?

Răspunsuri sugerate: pentru că ei pot vedea înainte, întreaga linie, ambele părți ale drumului și direcția spre care au plecat.

Cum se deplasează un robot pe un traseu marcat cu o culoare? Cum poate identifica traseul?

Utilizarea unui mediu de programare pentru implementarea algoritmilor

Elaborarea codului sursă pentru controlul robotului didactic virtual prin utilizarea și interpretarea datelor primite de la senzorii acestuia: evitarea obstacolelor, menținerea echilibrului, reacții specifice la detectarea luminii sau a identificării unui traseu marcat etc.

III. Lecții

LECȚIA URMĂRIREA LINIEI

Obiectivele lecției

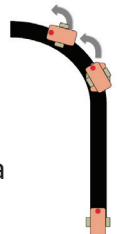
La sfârșitul lecției elevii vor ști:

- să cunoască modul în care oamenii și roboții se deplasează de-a lungul unei linii
- să programeze un robot să urmeze o linie utilizând senzorul de culoare
- să programeze un robot să urmeze o linie utilizând senzorul de lumină
- să programeze un robot să urmeze o linie până când un senzor este activat

Dacă vreți ca robotul din imaginea alăturată să urmărească linia utilizând senzorul de culoare ce comenzi îi puteți da?

Sugestie:

1. Dacă robotul "vede" culoarea neagră, să meargă drept.
2. Dacă robotul "vede" culoarea albă, să întoarcă la stânga pentru a reveni la linie.

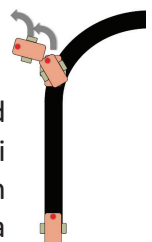


Sugestia este valabilă pentru robotul din imaginea alăturată?

Care sunt pașii pe care robotul trebuie să-i urmeze pentru a păstra traseul precizat?

Sugestie:

Deoarece linia are o grosime, putem programa robotul să urmeze linia astfel încât atunci când senzorul "vede" alb, robotul face o întoarcere (o rotire cu o roată oprită) spre negru și atunci când "vede" negru face o întoarcere spre alb. Acest algoritm este numit "urmărirea liniei în două etape" deoarece are doar două acțiuni. În plus, demonstrează modul în care combinația mișcărilor locale (spre dreapta sau spre stânga), conduce la o mișcare globală cu un sens foarte special, în acest caz urmărirea liniei.



ACTIVITATE PRACTICĂ

Programați robotul să se deplaseze pe un traseu dat.



Resurse necesare:

TIP ROBOT	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	De culoare	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Simple

REZOLVARE

SOLUȚIE 1 Programul pentru urmărirea liniei utilizând senzorul de culoare

```

+ start  show sensor data
repeat indefinitely
do
+ if  get colour  colour sensor Port 3  =  [black]
do
motor port B  on speed %  15
motor port C  on speed %  -3
else
motor port B  on speed %  -3
motor port C  on speed %  15

```

SOLUȚIE 2 Programul pentru urmărirea liniei utilizând senzorul de lumină

```

+ start  show sensor data
repeat indefinitely
do
+ if  15 <  get light  % colour sensor Port 3
do
motor port B  on speed %  10
stop motor port C  brake
else
motor port C  on speed %  10
stop motor port B  float

```

SOLUȚIE 3 Programul pentru urmărirea liniei până când un senzor este activat

```

+ start  show sensor data
repeat until  touch sensor Port 1  pressed?
do
+ if  get colour  colour sensor Port 3  =  [black]
do
motor port B  on speed %  15
motor port C  on speed %  -3
else
motor port B  on speed %  -3
motor port C  on speed %  15

```



SOLUȚIE 4 Programul optimizat cu variabile

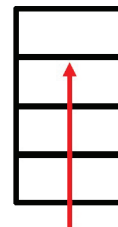
```

+ start show sensor data
- variable correction : Number ← 0.15
- variable speed : Number ← 15
- variable delta : Number ← 0
repeat until touch sensor Port 1 pressed?
do
set delta to 30 - get light % colour sensor Port 3
+ if 0 < delta
do
motor port B on speed % speed + delta × correction
motor port C on speed % speed
else
motor port B on speed % speed
motor port C on speed % speed - delta × correction
    
```



APLICAȚII PROPUSE

- Programați un robot să urmeze o linie pe o anumită distanță.
- Programați un robot să urmeze o linie până la tastarea ESC.
- Programați robotul să se oprească exact la a treia linie din imaginea alăturată. Trebuie să utilizați o buclă și un senzor! Ce senzor veți folosi?



LECȚIA SENZORUL TACTIL

Obiectivele lecției

La sfârșitul lecției elevii vor ști:

- să programeze un robot să urmeze o linie utilizând senzorul tactil
- să programeze un robot să urmeze o linie utilizând blocul așteaptă (Wait)

TEME DE DISCUȚII:

Când ai putea folosi senzorul tactil?

Comanda cea mai utilizată a senzorului este de amortizare. Amortizoarele sunt cele mai simple modalități de a interacționa cu mediul înconjurător; ele permit robotului să detecteze obstacolele în momentul în care le ating, aceștia schimbându-și orientarea corespunzător. Senzorul tactil poate avea trei aplicații: apăsat, eliberat sau de amortizare.

În OPEN ROBERTA LAB senzorul este practic ca un întrerupător True / False. Este util pentru programarea mișcării până când senzorul de atingere este apăsat / eliberat. De exemplu, dacă puneți un senzor tactil în fața robotului, îl puteți opri din mișcare. De asemenea, puteți porni sau opri programul când este apăsat un senzor tactil.



ACTIVITĂȚI PRACTICE

Programați robotul să pornească la atingere și să se oprească la detectarea culorii roșii.





Resurse necesare:

TIP ROBOT	SENZORI	PARAMETRI	SIMULATOR
Roberta EV3	Tactil	Parametrii trebuie să fie ajustați în funcție de scena aleasă.	Este recomandată scena Simple

REZOLVARE

Obstacol pentru ca senzorul tactil să fie apăsat la Start.



```

+ start
+ show sensor data
+ if touch sensor Port 1 pressed?
do
  drive forwards speed % 30
  + wait until get colour colour sensor Port 3 = 

```



APLICAȚII PROPUSE

- Programați robotul să se miște până când atinge un obstacol, apoi să se întoarcă și să pornească la dreapta cu 90 de grade.

LECȚIA SENZORUL ULTRASONIC

Obiectivele lecției

La sfârșitul lecției elevii vor ști:

- să utilizeze senzorul cu ultrasunete pentru a program robotul să păstreze distanță față de un obstacol
- să optimizeze codul

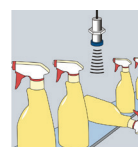
TEME DE DISCUȚII:

Dați exemple de utilizare a senzorului ultrasonic în viața reală.

Sugestii:

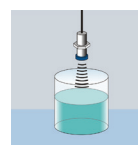
CONTROLUL CALITĂȚII

Detectează sticlele care sunt prea înalte sau prea mici, sau care sunt căzute pe o bandă transportoare.



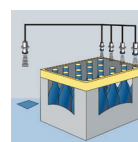
CONTROLUL NIVELULUI

Detectează nivelul de umplere a unui bazin de la câțiva milimetri până la 6 metri.



CONTROLUL ÎNCĂRCĂRII ÎN LADĂ

Verifică dacă lăzile sunt pline sau goale, sau dacă sticlele de plastic sunt goale pe o bandă transportoare.



Senzorul ultrasonic permite robotului să detecteze obiecte. El mai poate fi folosit pentru a evita obstacole, să măsoare distanțe și să detecteze mișcarea.

ACTIVITĂȚI PRACTICE

Scrieți un program pentru un robot care, folosind un senzor ultrasonic, să se întoarcă cu 180 de grade dacă este la 30 cm distanță față de un obstacol.

Resurse necesare:

TIP ROBOT	MOTOARE	SENZORI	SIMULATOR
Roberta EV3	Da	Ultrasonic	Este recomandată scena Drawing

REZOLVARE

```

+ start
show sensor data
repeat indefinitely
do
drive forwards speed % 30
+ if
get distance cm ultrasonic sensor Port 4 < 30
do
turn right speed % 30
degree 180
    
```

O problemă clasică de concurs în care este folosit senzorul ultrasonic este deplasarea constantă a robotului la o anumită distanță față de un perete. Pentru a realiza această cerință vom folosi un robot pe care senzorul ultrasonic să fie amplasat lateral.

APLICAȚII PROPUSE

- Citiți programul de mai jos și precizați reacțiile robotului.

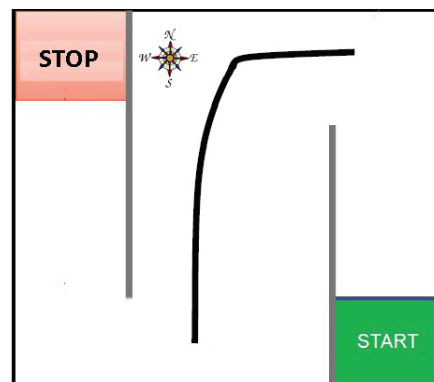
```

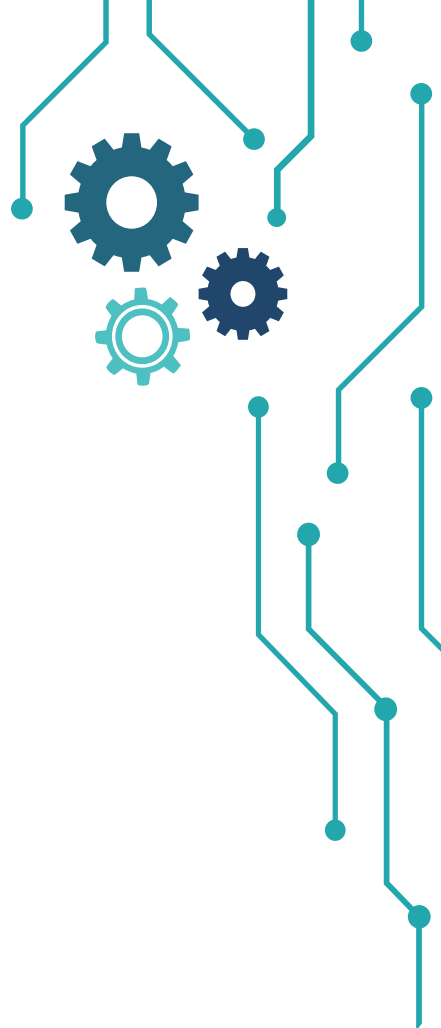
+ start
show sensor data
brick light
colour green
on
motor port B on speed % 100
repeat indefinitely
do
+ if
get distance cm ultrasonic sensor Port 4 ≤ 20
do
play frequency Hz 300
duration ms 100
+ if
get distance cm ultrasonic sensor Port 4 ≤ 10
do
stop motor port B brake
+ if
get distance cm ultrasonic sensor Port 4 ≤ 3
do
break out of loop
brick light off
    
```





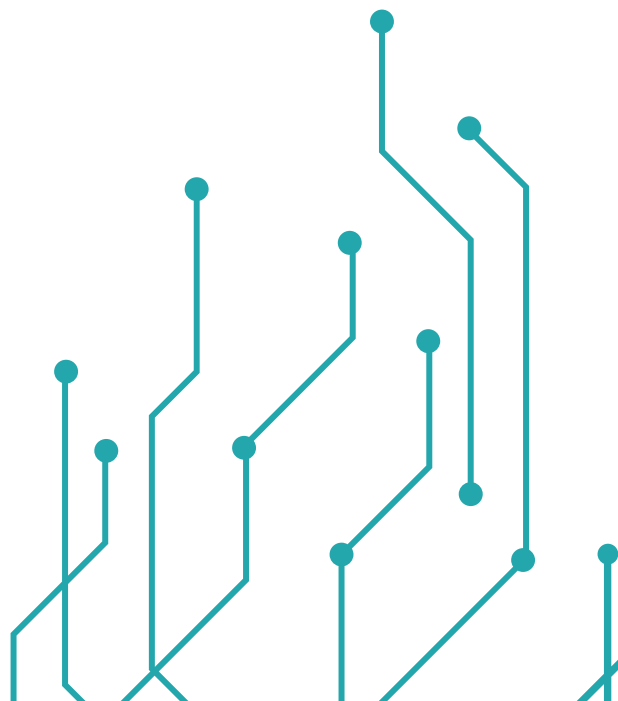
- Scrieți un program pentru un robot care urmează pașii de mai jos:
 - » Pleacă de la START
 - » Mergi spre Nord și atinge peretele Rotește spre Vest și NU atinge peretele.
 - » Mergi spre Sud și NU atinge peretele.
 - » Rotește și mergi la peretele din Vest.
 - » Rotește și mergi la STOP.





CAPITOLUL 4

MATERIALE DIDACTICE



IV. Materiale didactice

Învățarea orientată spre probleme nu are legătură cu succesul sau eșecul. Este vorba despre a fi activ învățând, construind și testând continuu ideile. Acest lucru trebuie înțeles și exersat de elevi.

Este important ca pe parcursul desfășurării activității la clasă profesorul să:

- evalueze performanța elevilor în fiecare etapă a procesului
- observe comportamentul, reacțiile și strategiile elevilor
- ofere feedback constructiv pentru a ajuta fiecare elev să progreseze
- adreseze întrebări elevilor despre ceea ce intenționează să pună în practică
- să ofere feedback la nivel de grup, când elevii lucrează în echipe
- încurajeze elevii să-și expună ideile utilizând texte, clipuri video, imagini, schițe sau alte medii creative
- ofere fiecărui elev libertatea de a-și alege modalitățile de exprimare

SISTEMUL DE MEDALII:



MEDALIE DE BRONZ

Elevul are puține cunoștințe teoretice, nu are capacitatea de a explica coerent o soluție sau de a pune în practică ceea ce știe.



MEDALIE DE ARGINT

Elevul este capabil să prezinte numai cunoștințe de bază (de ex., elemente de vocabular) dar nu poate aplica sau transmite mai departe ceea ce a învățat.



MEDALIE DE AUR

Elevul are un nivel de cunoștințe care îi permit să înțeleagă conținuturile și conceptele noi prezentate. Poate să pună în practică ceea ce a învățat. Nu poate să extindă raționamentul la o problemă nouă.



MEDALIE DE PLATINĂ

Elevul înțelege conținutul și conceptele prezentate, poate realiza sarcina de lucru, poate sintetiza și aplica conceptele și în alte situații.

ÎNȚELEGEREA CERINȚELOR

CERINȚA	SUFICIENT BRONZ	BINE ARGINT	FOARTE BINE AUR	EXCELENT PLATINĂ	NOTA
Descrieți problema în propriile cuvinte	Elevul nu poate descrie problema în propriile cuvinte.	Cu ajutor, elevul este capabil să descrie problema în propriile cuvinte.	Elevul este capabil să descrie problema în propriile cuvinte.	Elevul este capabil să descrie problema în propriile cuvinte și poate să împartă problema în subprobleme.	
Descrieți pașii necesari pentru obținerea soluției	Elevul nu poate descrie pașii soluției.	Cu ajutor, elevul este capabil să descrie pașii soluției.	Elevul este capabil să descrie pașii soluției.	Elevul este capabil să descrie în detaliu pașii soluției.	
Împărțiți problema în subprobleme	Elevul nu poate împărți problema în subprobleme.	Cu ajutor, elevul este capabil să împartă problema în subprobleme.	Elevul este capabil să împartă problema	Elevul este capabil să împartă problema în subprobleme și poate descrie legăturile dintre acestea.	



IDENTIFICAREA RESURSELOR

CERINȚA	SUFICIENT BRONZ	BINE ARGINT	FOARTE BINE AUR	EXCELENT PLATINĂ	NOTA
Descrieți resursele necesare pentru rezolvarea problemei (motoare, senzori, scenă,	Elevul nu poate să descrie resursele pe care le va folosi pentru rezolvarea problemei.	Cu ajutor, elevul poate să descrie resursele pe care le va folosi pentru rezolvarea problemei.	Elevul poate să descrie resursele pe care le va folosi pentru rezolvarea problemei și de ce.	Elevul poate să descrie, în detaliu, ce resurse va folosi pentru rezolvarea problemei și în ce mod.	
Recunoașteți și folosiți modelele sau noțiuni studiate	Elevul nu poate să recunoască modelele sau să folosească noțiunile	Cu ajutor, elevul poate să recunoască modelele sau să folosească noțiunile studiate anterior.	Elevul poate să recunoască modelele sau să folosească noțiunile studiate anterior.	Elevul poate să recunoască modelele și să folosească noțiunile studiate anterior în contexte noi.	

PROGRAMAREA SOLUȚIEI

CERINȚA	SUFICIENT BRONZ	BINE ARGINT	FOARTE BINE AUR	EXCELENT PLATINĂ	NOTA
Enumerati blocurile de programare necesare realizării programului	Elevul nu poate să descrie o listă de blocuri.	Cu ajutor, elevul poate să descrie o listă de blocuri.	Elevul poate să descrie o listă de blocuri.	Elevul poate să descrie o listă de acțiuni care să-l ajute să-și dezvolte programul.	
Descrieți succesiunea operațiilor din program.	Elevul nu poate să descrie succesiunea operațiilor.	Cu ajutor, elevul poate să descrie succesiunea operațiilor.	Elevul poate să descrie succesiunea operațiilor.	Elevul poate descrie programul oferind detalii despre fiecare etapă.	
Descrieți principiile de programare utilizate (ex. ieșire, intrări, evenimente, bucle, etc.	Elevul nu poate să descrie principiile de programare utilizate în soluție.	Cu ajutor, elevul poate să descrie principiile de programare utilizate în soluție.	Elevul poate să descrie principiile de programare utilizate în soluție.	Elevul poate să descrie principiile de programare utilizate în soluție cu detalii ce demonstrează înțelegerea principiilor de programare.	

VERIFICAREA SOLUȚIEI

CERINȚA	SUFICIENT BRONZ	BINE ARGINT	FOARTE BINE AUR	EXCELENT PLATINĂ	NOTA
Descrieți ce s-a întâmplat când s-a executat programul, și dacă este sau nu ceea ce v-ați așteptat.	Elevul nu poate să descrie ce s-a întâmplat.	Cu ajutor, elevul poate să descrie ce sa întâmplat comparativ cu cerința problemei.	Elevul este capabil să descrie ce s-a întâmplat comparativ cu cerința problemei.	Elevul este capabil să descrie ce s-a întâmplat, în comparație cu ce era de așteptat și găsește deja soluții.	
Descrieți ce problemele(erori) ați sesizat în rularea programului	Elevul nu poate să descrie cum problemele apărute	Cu ajutor, elevul poate să descrie problemele apărute	Elevul poate să descrie problemele apărute	Elevul poate descrie în detaliu cum a rezolvat problemele apărute.	
Descrieți unde credeți că apar problemele (erorile) în soluție	Elevul nu poate să descrie unde apare problema în soluție	Cu ajutor, elevul poate să descrie unde apare problema în soluție	Elevul poate să descrie unde apare problema în soluție	Elevul este capabil să descrie, în detaliu, modul în care eroarea afectează soluția.	
Descrieți câteva moduri în care ați încercat să rezolvați problema.	Elevul nu poate să descrie diferite moduri în care a încercat să rezolve problema.	Cu ajutor, elevul poate să descrie diferite moduri în care a încercat să rezolve problema.	Elevul poate să descrie diferite moduri în care a încercat să rezolve problema.	Elevul este capabil să descrie diferite moduri în care a încercat să rezolve problema și poate explica de ce aceste opțiuni nu au fost viabile.	



VERIFICAREA SOLUȚIEI

CERINȚA	SUFICIENT BRONZ	BINE ARGINT	FOARTE BINE AUR	EXCELENT PLATINĂ	NOTA
Descrieți cea mai importantă parte a soluției și dacă este sau nu	Elevul nu poate să descrie nicio parte din soluția găsită.	Cu ajutor, elevul poate să descrie cea mai importantă parte a soluției.	Elevul poate să descrie cea mai importantă parte a soluției, problemei.	Elevul poate să descrie soluția, evidențiind cea mai importantă parte a soluției.	
Descrieți cele mai importante detalii ale soluției date	Elevul nu poate să ofere detalii despre soluție	Cu ajutor, elevul poate să ofere detalii despre soluție	Elevul poate să ofere detalii despre soluție	Elevul poate să ofere detalii importante despre soluția găsită	
Descrieți cum soluția a atins obiectivul propus	Elevul nu poate să descrie modul în care soluția a atins obiectivul propus.	Cu ajutor, elevul poate să descrie modul în care soluția a atins obiectivul propus.	Elevul poate să descrie modul în care soluția a atins obiectivul propus.	Elevul poate să descrie, cu claritate modul în care soluția a atins obiectivul propus.	

ÎNVĂȚAREA PRIN JOC

Învățarea prin intermediul roboților este un mijloc de dobândire și utilizare a cunoștințelor prin acțiune, este o activitate de gândire orientată spre soluționarea unor probleme, spre găsirea căilor în vederea depășirii unor obstacole, stimulează procesul de instruire, îl adâncește și-l ameliorează. Învățarea cu roboți oferă un cadru propice pentru învățarea activă, participativă, stimulând în același timp inițiativa și creativitatea elevilor.

Învățarea cu roboți se poate concretiza în activități competiționale concepute pentru grupe de elevi sau individual. În timpul jocului fiecare elev acționează în ritmul său și rezolvă sarcina prin mijloacele operaționale de care dispune. Ca urmare, învățarea prin intermediul roboților activează elevii în procesul instructiv-educativ și, în aceeași măsură, vizează pe fiecare elev în parte, asigurând astfel un învățământ diferențiat.

Pe de altă parte, întrecerea este condiționată de procesul de instruire, calitatea și rezultatele fiind determinate de o bună pregătire anterioară.

Evaluarea finală a modului de robotică se poate realiza prin organizarea unui miniconcurs. Este de preferat ca în organizarea acestuia, să fie implicați elevii în stabilirea misiunii, realizarea scenei, a spațiului de concurs, în stabilirea regulamentului.

ORGANIZAREA UNUI CONCURS

Fiecare concurs are un regulament propriu în funcție de obiectivele propuse, de categoriile de vârstă ale concurenților, de tipurile de roboți folosiți, tipurile de probleme rezolvate, etc.

În vederea participării la un concurs de robotică, pregătirea echipei trebuie să se facă având în vedere următoarele elemente:

- Fiecare echipă trebuie să studieze cu atenție regulamentul jocului.
- Fiecare echipă va avea un antrenor: elev de liceu, student sau profesor.
- Contestarea acordării punctajelor se face conform regulamentului. (Învățați elevii să se comporte politicos.)
- Terenul de joc din concurs este întotdeauna puțin diferit de cel pe care v-ați antrenat (pot interveni factori ambientali precum zgomot, lumină, etc.), de aceea toate concursurile prevăd o etapă preliminară de programare și testare a robotului pe suprafața de concurs.



Termeni utilizați în concursurile de robotică

Misiunea este o cerință pe care robotul o va îndeplini pentru acumularea de puncte.

Meciul este o sesiune de întreceri. Într-un turneu, două echipe vor concura simultan într-un meci. Există întreceri în care concurează câte 2 echipe împotriva altor 2. Premiul este dat celui cu cele mai multe meciuri câștigate.

Suprafața de joc (scena) este un spațiu amenajat pentru desfășurarea meciurilor. Aceasta include suprafața de joc (ex. masă, ring), pereții interiori, marginile și obstacolele. În interiorul fiecărei scene există o bază.

Baza este locul din care robotul începe fiecare meci și zona în care concurenții pot atinge robotul fără penalizare.

Standul este locul unde echipa poate depozita și asambla piesele pentru o anumită misiune.

Puteți studia, ca model, concursurile existente în spațiul competițional românesc pentru a vă inspira în realizarea propriului concurs.



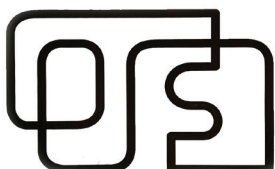
ACTIVITĂȚI PRACTICE

Studiați concursurile **RoboSmart Junior** și **Concursul FIRST 2016**, prezentate mai jos. Comentați diferențele și asemănările dintre misiunile celor două concursuri. Enunțați o misiune pentru un concurs nou. Desenați scena pentru misiunea enunțată mai sus. Stabiliți un regulament pentru misiunea enunțată. Dați un nume concursului vostru.

Concursul RoboSmart Junior

Misiune: Realizați un tur de circuit fără erori.

Scenă:



Regulament:

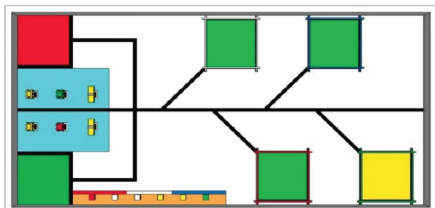
- » Fiecare echipă are la dispoziție 5 încercări cronometrate dintre care se ia cel mai bun timp obținut.
- » Fiecare echipă are la dispoziție 30 de minute pentru a pregăti programul pentru testele oficiale, după care echipele se vor prezenta la baza. În timpul testelor oficiale, echipele au dreptul să optimizeze programul până sunt chemate la pistă.
- » Dacă robotul pierde linia, acesta trebuie să reia singur traseul dintr-un punct atins anterior sau chiar din punctul de unde a pierdut linia. Dacă nu poate relua singur traseul, atunci punctul în care a pierdut linia este considerat cel mai înaintat punct în care a ajuns robotul.
- » La un test oficial, fiecare echipă are dreptul la 2 încercări. Dacă robotul nu ajunge la capăt din 2 încercări atunci este notat cel mai înaintat punct în care a ajuns la acel test.
- » La finalul fiecărui test oficial echipele au la dispoziție 10 minute pentru a face optimizări la program.



Concursul FIRST 2016

Misiune: Misiunea robotului este de a reduce amprenta de carbon a unei companii la zero. Amprenta de carbon a unei companii este cantitatea de dioxid de carbon emisă de procesele industriale ale companiei. Robotul trebuie să instaleze panouri solare și să planteze copacii potriviți în diferitele zone de plantare pentru a contrabalansa emisiile de carbon ale companiei.

Scenă:



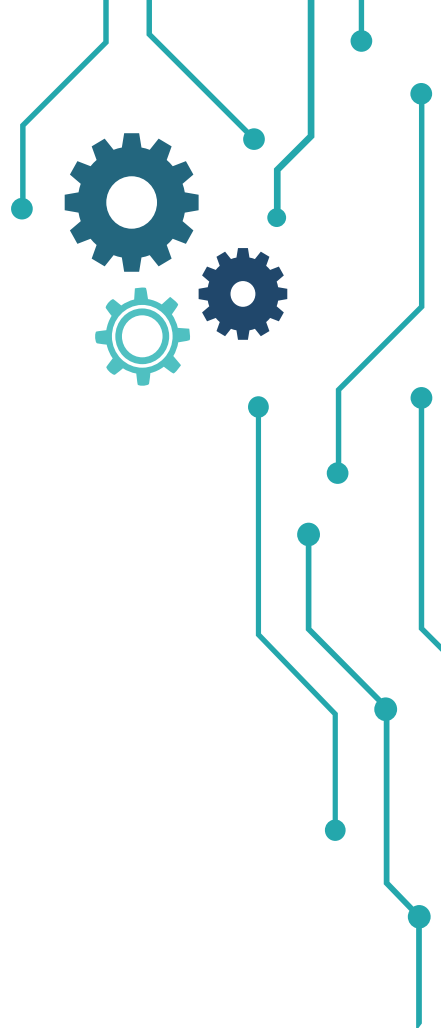
Regulament:

- » Competiția are un număr de runde, timp de asamblare (150 minute), programare și testare. Competiția WRO Romania 2016 va avea un număr de 2 runde (evaluări oficiale), Runda 1 - timp de asamblare, programare și testare 120 minute, respectiv Runda a 2-a - timp de modificare, programare și testare 60 minute.
- » Concurenții nu pot asambla Robotul în afara timpilor alocați pentru asamblare/modificare, programare și testare, acordate înainte de fiecare rundă/evaluare.
- » Concurenții pot începe asamblarea robotului din momentul în care STARTUL a fost anunțat oficial în cadrul evenimentului. Programarea și testarea robotului poate începe după 45 minute de la semnalul de START al competiției.
- » Atunci când timpul de asamblare, programare și testare a expirat, echipele trebuie să își poziționeze Robotul în zona desemnată pentru inspecție ("zona de carantina"). Arbitrii vor evalua dacă Robotul se conformează tuturor reglementărilor competiției (dimensiune, componente, sisteme de comunicații, programare, etc).
- » Dacă în timpul inspecției se constată o încălcare a regulilor, arbitrul va acorda echipei un timp de 3 (trei) minute pentru corectarea acesteia. Echipa nu poate participa la runda/concurs dacă nu se remediază situația în timpul alocat pentru corecție.
- » Robotul va avea un singur program executabil cu numele "run2016".
- » Robotul are la dispoziție 2 minute pentru a termina proba. Cronometrul pornește atunci când arbitrii dau semnalul de start. Robotul trebuie poziționat în zona de start .
- » Când participanții au terminat ajustările finale, arbitrii vor da semnalul pentru a porni brick-ul NXT/ EV3 și pentru a selecta un program (fără a rula programul). După deschiderea robotului și selectarea programului, arbitrii vor întreba echipa care este modul de rulare a robotului. Sunt posibile două variante: a. Robotul se pune în mișcare imediat după ce rulează programul b. Robotul se pune în mișcare după apăsarea butonului central, alte butoane și senzori nu pot fi utilizate pentru start. Dacă este indicată opțiunea a) arbitrul va da semnalul de start și un membru al echipei va rula programul. Dacă este folosită varianta b) un membru al echipei rulează programul și așteaptă startul. La acest moment nu mai sunt acceptate modificări ale poziției sau părților componente ale robotului. Arbitrul va da semnalul de start și un membru al echipei va apăsa butonul central pentru a porni robotul.
- » Pe parcursul desfășurării rundelor de concurs (evaluări), deciziile luate de arbitri sunt incontestabile
- » Runda/Evaluarea și timpul se vor opri dacă: a. Timpul de 2 minute alocat probei s-a terminat b. Un membru al echipei atinge robotul în timpul probei c. Robotul a părăsit complet suprafața de competiție d. Încălcarea oricărei reguli și reglementari din prezentul regulament e. Misiunea a fost îndeplinită.
- » Pentru fiecare echipă, arbitrii vor calcula punctajul la încheierea fiecărei runde/evaluări. Echipa trebuie să verifice pe loc punctajul acordat și să semneze fișa de jurizare la finalul fiecărei runde.



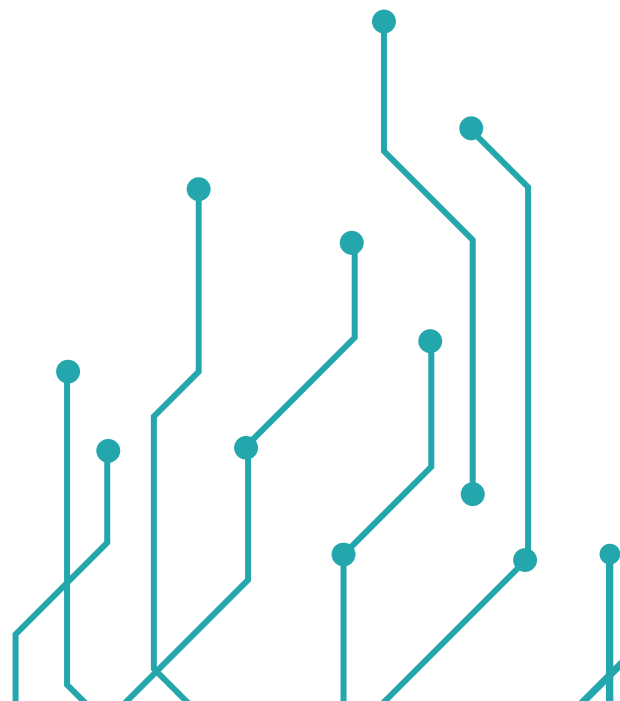
- » Clasamentul echipelor este decis în funcție de cel mai bun scor obținut în competiție. Dacă echipele au același scor, departajarea acestora se face în funcție de timpul obținut la rundă. Dacă echipele rămân în continuare la egalitate, departajarea acestora va fi făcută în funcție de punctajul obținut în rundele anterioare. În cazul în care există echipe care obțin același punctaj și același scor la ambele runde din competiție, acestea vor fi departajate printr-o proba suplimentară (care poate fi diferită de probele anterioare).





CAPITOLUL 5

ROBOTICA COMPETIȚIONALĂ



V. Robotica competițională

În acest moment, robotica competițională este practică în cluburi, cercuri, palate, centre de excelență, ONG-uri. Activitatea competițională presupune cunoștințe aprofundate, soluții inovative, multe ore de muncă, teste încercări. Competițiile recunoscute de MEN în calendarele competiționale preuniversitare și universitare sunt pentru toate categoriile de vârstă (sub 12 ani, 13-15 ani, 16-19 ani, peste 19 ani).

De regulă, la începutul anului competițional se anunță o temă, o problemă pentru care o echipă se pregătește pe tot parcursul anului. Temele variază de la întrecerile clasice precum urmărirea liniei, labirint cu linii, labirint cu pereți, Sumo, LaserMaxx (plasarea a patru roboți autonomi pe un ring circular, unde scopul este evitarea ieșirii din teren și lovirea cu laserul a adversarilor) până la teme tehnice, ecologice, sociale.

COMPETIȚII DE ROBOTICĂ ORGANIZATE ÎN ROMÂNIA

FIRST® LEGO® League

FIRST® LEGO® League este un program dedicat copiilor și tinerilor (9 - 16 ani), ce are ca scop introducerea acestora în lumea fascinantă a științei și tehnologiei, prin intermediul unei abordări sportive.

Obiectivele FIRST® LEGO® League sunt de a entuziasma copiii și tinerii cu privire la știința și tehnologie, familiariza participanții cu ideea de spirit de echipă, încuraja copiii și tinerii să rezolve sarcini complexe într-un mod creativ.

Componenta de baza a FIRST® LEGO® League este reprezentată de un turneu de robotica organizat într-o atmosferă veselă, în cadrul căruia copiii și tinerii (12 – 19 ani) sunt provocați să rezolve o “misiune” complicată, cu ajutorul unui robot. Copiii, organizați în echipe, vor cerceta un subiect dat, vor planifica și realiza programarea și testarea unui robot autonom pentru a rezolva misiunea dată. Echipele FIRST® LEGO® League au posibilitatea de a experimenta toate etapele din cadrul procesului de dezvoltare al unui produs real: rezolvarea unei probleme sub presiunea timpului, cu resurse insuficiente și a unor concurenți necunoscuți. Concursul are și etapă internațională.

www.firstlegoleague.ro

World Robotics Olympiad™

World Robot Olympiad™ este un concurs dedicat științei și tehnicii, care dorește antrenarea unui număr cât mai mare de tineri din întreaga lume, cu scopul de a le stimula și dezvolta creativitatea și spiritul inovativ. WRO este o competiție pentru echipe. Pentru a participa la fiecare Categorie din competiție elevii trebuie să lucreze în echipe. O echipă este formată dintr-un antrenor și doi sau trei membri.

Obiective

- Furnizează elevilor, cu vârste de până la 19 ani, oportunitatea de a-și extinde orizonturile prin intermediul sistemelor robotice existente în școli;
- Introducerea conceptului de științe moderne în activitățile curriculare din instituțiile de învățământ;
- Oferă oportunitatea de a promova gândirea creativă, de a îmbunătăți abilitățile de comunicare, de a consolida capacitatea de a dobândi noi cunoștințe, cele mai relevante pentru educație;
- Reunește tineri din întreaga lume cu scopul de a le dezvolta creativitatea și abilitățile practice;



- Extinde viziunea tinerilor în domeniul științei și tehnicii prin îmbunătățirea procesului de învățare, încurajându-i în a deveni viitori ingineri, cercetători sau inventatori.

Concursul are și etapă internațională.

www.wroromania.ro

BDR First Tech Challenge

Este cel mai mare program de robotică pentru elevi de liceu desfășurat în România începând cu anul școlar 2016-2017.

În fiecare sezon competițional, participă peste 1500 de elevi și 300 de mentori-profesori, din peste 50 de orașe din România. Programul de robotică include și sesiuni de training, workshops tehnice și de soft-skills, meciuri amicale și evenimente demonstrative.

Echipele sunt susținute cu kit-uri de robotică, truse de scule și imprimante 3D de Asociația Nație Prin Educație și BRD FIRST Tech Challenge România.

Programul este adus în România cu licență de la FIRST USA (For Inspiration and Recognition in Science and Technology) și are un format diferit față de alte concursuri de robotică desfășurate în România. Astfel, robotul este construit de fiecare echipă din piesele tetrax din kit-urile primite sau pe care le poate achiziționa suplimentar din fonduri proprii, în funcție de sponsorii atrași, sau le pot construi folosind imprimanta 3D, respectând anumite specificații tehnice.

În cadrul concursului, robotul acționează timp de 2 minute și 30 de secunde, din care, în primele 30 se deplasează autonom, pe baza unui program, după care cu ajutorul joystick-ului robotul este controlat de către membrii echipei. În ultimele 30 de secunde, robotul trebuie să realizeze o sarcină anume, de regulă cea mai dificilă. Meciurile au 4 combatanți, 2 echipe roșii și 2 albastre care formează alianțe. Meciul este câștigat de perechea care însumează cele mai multe puncte. Numărul de meciuri câștigate stabilește locul pe podium.

Concursul are și etapă internațională, în SUA, la Campionatul Mondial de Robotică.

www.natieprineducatie.ro

Robochallenge

Robochallenge este o competiție de robotică care se adresează în special studenților pasionați de partea practică a unui proiect, dar și celor dornici să învețe mai mult. Concursul este adresat atât începătorilor, cât și avansaților, dar și pasionaților de robotică.

Concursul vine ca o completare la partea teoretică ce se învață în cadrul facultății, fiind modul prin care studenții pot face ceva plăcut și util folosind cunoștințele teoretice.

Echipele primesc un regulament și pe baza lui își construiesc roboții de la 0, folosind componente și materiale aflate la îndemâna lor. Fiecare echipă are propria viziune și abordare, și datorită acestui fapt apar surprize în fiecare an, care arată inventivitatea acestora.

Concursul se adresează oricărui pasionat/ă de robotică, ne-existând nici o limită de vârstă, dar există două categorii, dintre cele 13 existente, dedicate exclusiv copiilor.

www.robochallenge.ro





CONCURSURI ORGANIZATE DE PALATELE COPIILOR:

Trofeul de Robotică Robotor, Palatul Copiilor Orșova

Trofeul Internațional de Robotică ROBOTOR promovează robotica în rândul elevilor (fete și băieți) din România și Europa, precum și a publicului larg, prin intermediul competițiilor de robotică.

Robotor a fost inclus în fiecare an, începând cu anul 2008, în Calendarul Activităților Educative al Ministerului Educației Naționale ca proiect regional, național sau internațional. Au fost organizate o ediție regională (2008), cinci ediții naționale (2009, 2010, and 2012 – 2014), și patru internaționale (2011, 2015 – 2017). Edițiile 2013 și 2014 au fost organizate în colaborare cu Palatul Copiilor Botoșani.

www.robotor.ro

Cupa Chindiei, Palatul Copiilor Târgoviște

Scopul Concursului Național de Robotică „Cupa Chindiei” este acela de a stimula dezvoltarea activităților tehnico-aplicative în domeniul roboticii din palate și cluburi ale elevilor, școli și licee, cluburi, universități; de a dezvolta la elevi și tineri, interesul față de inteligența artificială – Robotică, Mecatronică și limbaje de programare. Cupa Chindiei figurează în Calendarul Activităților Educative Naționale al Ministerului Educației Naționale (CAEN).

La concurs poate participa o echipă formată din 2-4 membrii, cu remarca că aceeași echipă nu poate participa la două categorii separate, iar un elev nu poate face parte din două echipe .

Există două categorii ale concursului: Categoria A, unde participă roboți construiți în totalitate (fac excepție, motoarele, roțile, acumulatorii) și nu se admit subansambluri comerciale, Categoria B unde participă roboți ce folosesc module industriale și Categoria C unde participă roboți de orice fel.

Concursul interjudețean “Să descoperim mecatronica și robotica”, Facultatea de Automatică, Calculatoare și Electronică Craiova, Departamentul de Mecatronică și Robotică

Este adresat elevilor din clasele IX-XII din 2015 și are două secțiuni: Mecatronică și Robotică. Fiecare școală poate participa cu 2 echipe de câte 3 elevi.





DESPRE ASOCIAȚIA E-CIVIS

Asociația E-Civis este o organizație non-guvernamentală înființată în anul 2009.

www.e-civis.eu

Misiunea organizației E-Civis este de a promova valorile democratice la nivel național și internațional, în special prin intermediul noilor tehnologii. E-Civis are în vedere informarea cu privire la valorile democratice și stimularea actorilor sociali de a le promova și pune în practică la nivel național și internațional. E-Civis organizează și susține evenimente de informare dedicate tinerilor și activități de educație non-formală pe domeniile de expertiză (educație civică, codare, robotică), atât în mediul rural cât și urban.

Proiecte ale organizației privind educația digitală:

Săptămâna Europeană a Roboticii

www.saptamanaroboticii.ro

Săptămâna Europeană a Roboticii este o inițiativă a rețelei europene EURobotics, începută în 2011, ce are drept scop promovarea industriei roboticii în țările Uniunii Europene.



Săptămâna Europeană a Roboticii oferă o săptămână de activități variate pentru publicul larg din Europa, pentru a sublinia importanța acestei industrii în cele mai variate domenii. De asemenea, prin aceste activități se dorește stimularea studenților de orice vârstă să urmeze o educație tehnologică, fie ea programare, robotică, inginerie sau alte domenii.

Oricine poate organiza un eveniment în cadrul acestei inițiative europene ce are loc în fiecare an, la jumătatea lunii noiembrie.

Girls Champions in 3D

www.girlschampions.com

#GirlsChampionsin3D este un proiect de inovare socială în domeniul noilor tehnologii.

Scopul #GirlsChampionsin3D este de a crea o piață a muncii mai incluzive, în special în domeniul tehnologiei pentru femeile din întreaga lume. Modelarea și imprimarea 3D ca un domeniu IT al femeilor.

Proiectul este derulat în parteneriat cu ECDL ROMÂNIA, care a instruit și certificat 22 de studente din București până în acest moment.

RoboHub

Robo Hub este dezvoltat în cadrul proiectului Everyone Digital:

<http://e-civis.eu/alt-viitor-everyone-digital/>



RoboHub este un spațiu dedicat comunității celor pasionați de robotică din București, dar și din țară. RoboHub oferă cursuri gratuite și subvenționate pentru copiii ai căror părinți nu își permit costul unui curs de robotică.

La RoboHub au loc și cursuri de introducere în noi tehnologii conexe roboticii, precum Machine Learning, Inteligență Artificială, HoloLens, Robotic Operating System, dar și dezbateri privind impactul noilor tehnologii asupra societății noastre.





Forumul European al Roboticii – ediția a X-a

www.eu-robotics.net/robotics_forum/

Forumul European al Roboticii este locul de întâlnire anual al industriei de robotică și va reuni peste 1.000 de cercetători, ingineri, manageri și un număr din ce în ce mai mare de antreprenori și oameni de afaceri de pe continent. Evenimentul va avea loc între 20 și 22 martie 2019, în București.

Forumul European al Roboticii este organizat de euRobotics sub umbrela SPARC, un parteneriat public privat pentru dezvoltarea roboticii în Europa. ERF 2018 este găzduit de Asociația E-Civis, în colaborare cu Universitatea Politehnică din București și Universitatea București, Facultatea de Matematică și Informatică.

DESPRE AUTOR

Marilena Oprea este profesor de informatică titular la Colegiul Național Unirea Focșani din 1993.

A publicat, fiind co-autor, cărțile de specialitate **BPDQ Culegerea de probleme de informatică**, editura Pallas, Focșani, 2006, **Ghid Metodic BACALAUREAT 2007 informatica**, Editura Gil, București, 2007, **Tehnici de optimizare, culegere de probleme**, Editura Infodata, Cluj, 2008, **Prelucrarea Imaginilor**, editura MagicPrint, Onești, 2009, **Matematica de gimnaziu în concursurile vrâncene**, editura Andrew, Focșani, 2011, **Tehnologia Informațiilor și Comunicațiilor** pentru clasa a VI-a, Indrumar, CNEE 2012, **ABC-UL PROGRAMĂRII**, editată online ISBN 978-973-0-23485-5, 2016.

Este referent pentru publicațiile **Manual Informatica si TIC**, clasa a V-a, editura Intuitex, 2017 și **Studii Informatic**, editura CCD Bacău, 2015.

A scris o serie de articole, în revistele de specialitate, care reflectă preocuparea pentru transformările din sistemul educațional românesc: „Elemente transformatoare ale educației digitale în 2015”, „Educația eSafety”, „Disciplinele informatice-scurtă analiză”, „Un an de robotică la CNU”, „Pasionații de robotică”. A publicat 600 de itemi și 25 de teste în platforma educațională INSAM.

Ca profesor s-a implicat în pregătirea elevilor pentru olimpiadele și concursurile școlare, pentru examenele naționale de bacalaureat și atestare profesională la informatică. Este formator CISCO din 2011. Pregătirea susținută a elevilor s-a concretizat în peste 150 de premii obținute cu elevii la olimpiadele și concursurile județene și naționale și 2 premii internaționale - bronz la CEOI și la IOI în 2006.

În perioada 2007-2018, a făcut parte din comisiile centrale a olimpiadelor de Informatică și de Tehnologia Informației și a contribuit la elaborarea subiectelor unice pentru etapele județene și naționale.

Din 2016 face parte din Comisia Națională de specialitate la disciplina Tehnologia Informației și Comunicațiilor.

Recunoașterea profesionalismului didactic a dus la cooptarea doamnei prof. Marilena Oprea în grupuri de lucru naționale implicate în reformarea curriculară precum grupul CNEE de elaborare a subiectelor pentru examenul de bacalaureat național, grupul de lucru pentru evaluarea manualelor școlare sau grupul de lucru pentru elaborarea programelor școlare de gimnaziu finalizate prin OMEN nr. 3393 / 28.02.2017.

În calitatea de inspector școlar, în perioada 2008-2016, Marilena Oprea a inițiat activități menite să promoveze programarea în rândul elevilor înființând și coordonând, la nivel județean, **Centrul de excelență de Robotică Educațională** și **Centrul de pregătire a elevilor capabili de performanță la matematică și informatică**.





În categoria inițiativelor care au avut în vedere creșterea impactului educației folosind noile tehnologii, se încadrează o serie de instruirii realizate cu profesorii din județul Vrancea prin programe și proiecte precum **Competențe TIC în curriculum școlar, Google for Education, E-scoala, certificarea eSafety, SigurInfo.**

Din 8 septembrie 1998 este președinta Asociației pentru Acces la Informație, Educație, Cultură și Tehnologie "Proiect".

Din 2017 coordonează colectivul de redacție al revistei ROBOTICA .





Bibliografie și resurse

OMEN nr. 3393 / 28.02.2017, Anexa nr. 2 OMEN nr. 3393 / 28.02.2017

Seymour Papert, Mindstorms: Children, Computers, and Powerful Ideas,

Beth Gardiner, „Adding Coding to the Curriculum”, NY Times

Mitchel Resnick, John Maloney, i in., „Scratch: Programming for All”

Seymour Papert, „Educational Computing: How Are We Doing?” The Next Web

Open Roberta Developer Website

Blockly Website

Dokumentation NEPO

http://microbit.org/en/2018-01-19-train_the_trainer/

<http://microbit.org/teach/iet/>

<https://www.youtube.com/watch?v=uHll1-7zDR8>

<http://www.mil.ufl.edu/~chrisarnold/components/microcontrollerBoard/AVR/avr/lib/>

<http://bricxcc.sourceforge.net/nbc/>

<http://www.embedded.com/2000/0010/0010feat3.htm>

www.coppeliarobotics.com

www.robotvirtualworlds.com

www.virtualroboticstoolkit.com

www.raspberrypi.org

https://en.wikipedia.org/wiki/Open_Roberta

www.legoengineering.com/ev3-sensors/

www.facebook.com/groups/LEGOengineering/videos/





INDEX

<ul style="list-style-type: none"> Edit11 <ul style="list-style-type: none"> run on robot 1 run in simulation 12 new 12 my programs 12 example programs 12 save 12 save as 12 Show code 12 Create program link 12 Export program 12 Import program 13 Help menu 13 <ul style="list-style-type: none"> general help 13 FAQ 13 about 13 User 14 <ul style="list-style-type: none"> login 14 logout 14 change 14 delete user 14 state information 14 Language menu 14 Galeria Open Roberta Lab 14 Simulatorul Open Roberta Sim 13 Butoanele de control în Open Roberta Sim 15 Pornirea unui program pe robot 16 Brick-ul Robotului EV3 17 Robotul 2D Roberta EV3 18 Robotul 3D EV3 18 Blocurile NEPO 20 Meniul contextual 21 Categorii de blocuri 22 <ul style="list-style-type: none"> 1. Categoria Action (Acțiune) 22 <ul style="list-style-type: none"> Activități practice 23 Aplicații propuse 26 2. Categoria Control 26 <ul style="list-style-type: none"> Activități practice 26 Aplicații propuse 29 3. Categoria Logic 29 <ul style="list-style-type: none"> Activități practice 29 Aplicații propuse 30 4. Categoria Matematică (Math) 30 <ul style="list-style-type: none"> Activități practice 31 Aplicații propuse 33 5. Categoria Text 33 <ul style="list-style-type: none"> Activități practice 33 Aplicații propuse 34 6. Categoria Variabile (Variables) 34 <ul style="list-style-type: none"> Activități practice 35 Aplicații propuse 35 7. Categoria Liste (Lists) 35 <ul style="list-style-type: none"> Activități practice 36 Aplicații propuse 36 8. Categoria Funcții (Functions) 36 9. Categoria Mesaje (Messages) 37 10. Categoria Senzori (Senzori) 37 11. Categoria Culori (Colors) 38 <ul style="list-style-type: none"> Senzorul de culoare/lumină 38 Senzorul tactil 38 Senzorul ultrasonic 39 Activități practice 39 Aplicații propuse 40 Lecția Urmărirea liniei 42 <ul style="list-style-type: none"> Activități practice 42 Aplicații propuse 44 Lecția Senzorul tactil 44 <ul style="list-style-type: none"> Activități practice 44 Aplicații propuse 45 Lecția Senzorul ultrasonic 45 <ul style="list-style-type: none"> Activități practice 46 Aplicații propuse 46 Materiale didactice 49 Învățarea prin joc 51 Organizarea unui concurs 51 Robotica competițională 56

