

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

PROGRAMOVANIE ROBOTOV POMOCOU
STAVOVÝCH AUTOMATOV
BAKALÁRSKA PRÁCA

2024

TOMÁŠ VIKISZÁLY

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

PROGRAMOVANIE ROBOTOV POMOCOU
STAVOVÝCH AUTOMATOV
BAKALÁRSKA PRÁCA

Študijný program: Aplikovaná informatika
Študijný odbor: Aplikovaná informatika
Školiace pracovisko: Katedra aplikovanej informatiky
Školiteľ: Mgr. Pavel Petrovič, PhD.

Bratislava, 2024
Tomáš Vikiszály



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:

Študijný program:

Študijný odbor:

Typ záverečnej práce:

Jazyk záverečnej práce:

Sekundárny jazyk:

Názov:

Anotácia:

Vedúci:

Katedra:

Vedúci katedry:

Dátum zadania:

Dátum schválenia:

garant študijného programu

.....
študent

.....
vedúci práce

Pod'akovanie:

Abstrakt

Kľúčové slová: LEGO Education Spike Prime, stavové automaty, programovanie robotov

Abstract

Keywords: LEGO Education Spike Prime, state machines, robot programming

Obsah

Úvod	1
1 Stavebnica LEGO Spike Prime	3
1.1 Obsah balenia stavebnice	3
1.2 Programátorské rozhranie a pripojenie sa k riadiacej jednotke	3
1.3 Vzorové programy	6
1.3.1 Pohyb do štvorca	6
1.3.2 Sledovanie čiary	7
1.3.3 Sledovanie objektu	9
2 Špecifikácia programu	11
2.1 Špecifikácia platformy a jazyka	11
2.2 Komunikácia programu s robotom	11
2.3 Rozhranie programu	11
2.4 Stavový automat	11
2.5 Stavy	12
2.6 Prechody medzi stavmi	12
2.7 Inicializačný stav	12
2.8 Konečný stav	12
2.9 Superprechod	12
2.10 Premenné	13
2.11 Knižnica	13
Záver	15
Príloha A	19
Príloha B	21

Zoznam obrázkov

1.1	Ukážka balenia stavebnice	4
1.2	Jednoduchý robot	6
1.3	Program pohybu robota do štvorca	7
1.4	Jednoduchý robot so senzorom	8
1.5	Program pohybu robota po čiare	8
1.6	Jednoduchý robot s ultrazvukovým senzorom	9
1.7	Program pohybu robota ktorý sleduje objekt pred sebou	10

Zoznam tabuliek

Úvod

...

Kapitola 1

Stavebnica LEGO Spike Prime

Pre náš výskum a implementáciu využívame stavebnicu LEGO Education SPIKE prime. Developer nám ponúka možnosti využívať webový editor kódu alebo aplikáciu na stiahnutie, ktorá obsahuje rovnako všetky pomôcky na programovanie ako jej internetová verzia. Na začiatku si predstavíme obsah jedného balenia stavebnice, programátorské rozhranie stavebnice ako aj jednoduché skripty v tomto jazyku.

1.1 Obsah balenia stavebnice

Každá stavebnica predávaná firmou LEGO na ich webovej stránke pozostáva z týchto hlavných komponentov (viď obr. 1.1):

- riadiacej jednotky v ktorej je vstavný gyroskop, reproduktor, dva tlačidlá a svetelný displej rozmerov 5x5 štvorcov (vrátane nabíjacieho káblu),
- dvoch malých a jedným väčším servomotorom,
- farebný senzor,
- infračervený senzor,
- dotykový senzor,
- kolieska a spojovacie materiály,
- a ďalšie súčiastky potrebné na realizáciu stavby robota.

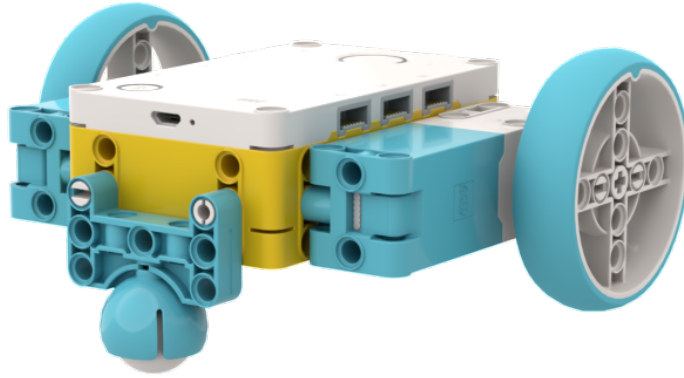
1.2 Programátorské rozhranie a pripojenie sa k riadiacej jednotke

Po zvolení si možnosti nového projektu dostaneme na výber z dvoch možností, ktoré sú 'ICON BLOCKS' a 'WORLD BLOCKS'. Pre naše účely budeme využívať druhú mož-

1.2. PROGRAMATORSKÉ ROZHRANIE A PRIPOJENIE SA K RIADIACEJ JEDNOTKE5

- zvukov:
 - zapínanie a vypínanie prehrávania zvukov
 - ovládanie dĺžky prehrávania zvukov
 - ovládanie hlasitosti
- akcii:
 - možnosť spúšťania nového procesu ako reakciu na zmenu alebo vonkajší vplyv
 - možnosť spustiť nový proces
- ovládania:
 - možnosť pozastavenia behu programu na určitý čas alebo kým nebude splnená nejaká podmienka
 - opakovanie určitého bloku príkazov
 - vetvenie programu cez podmienky
 - vypnutie všetkých alebo niektorých procesov
- senzorov:
 - snímanie farieb, zvukov, tlaku, polohy alebo vstupu tlačidiel
 - ovládanie a čítanie času
- operácii:
 - riešiť rôzne číselné alebo logické operácie
- premenných a vlastných blokov, ktoré si môžeme ľubovoľne tvoriť pokiaľ to nebude presahovať pamäť riadiacej jednotky.

Pre výber komponentu z ponuky ho ťaháme a pustíme ho na mieste kde ho chceme vložiť. Prostredie je interaktívne a pri nadídení jedným komponentom nad druhý sa nám sivým rámčekom naznačuje, ako bude implemenotvaný po pustení. Pre úspešné prepojenie počítača a riadiacej jednotky máme dve možnosti: káblové pripojenie alebo bluetooth. V oboch prípadoch musíme riadiacu jednotku zapnúť veľkým kruhovým tlačidlom v strednej spodnej časti a kliknutím žltej ikonky obrázka riadiacej jednotky v ľavom hornom rohu obrazovky počítača a následne si zvolíme typ spojenia. Pre káblové spojenie vyberieme z ponuky pravú možnosť s bielym tlačidlom, zapojíme jeden koniec kábla do riadiacej jednotky a druhý do počítača. V prípade bezkáblového spojenia si vyberieme ľavú možnosť so zeleným tlačidlom, kedy nám prostredie ponúkne všetky



Obr. 1.2: Na obrázku je možné vidieť robota, ktorého budeme požívať v podkapitole 1.3.1. Obrázok bol vytvorený pomocou stud.io aplikácie.

riadiace jednotky v dosahu. Ak používame aplikáciu, tak tento zoznam bude v pravej časti obrazovky počítača a v rámci webového rozhrania to najdeme ako vyskakovacie okno prehliadača. Po pripojení bezdrôtovým spôsobom zelené tlačidlo prestane blikať a začne svietiť zeleným svetlom. Kedykoľvek vieme skontrolovať stav pripojených komponentov v ľavej hornej časti obrazovky napravo od pripájacieho tlačidla a po kliknutí na tlačidlo vieme získať informáciu aj o percentuálnom stave baterie.

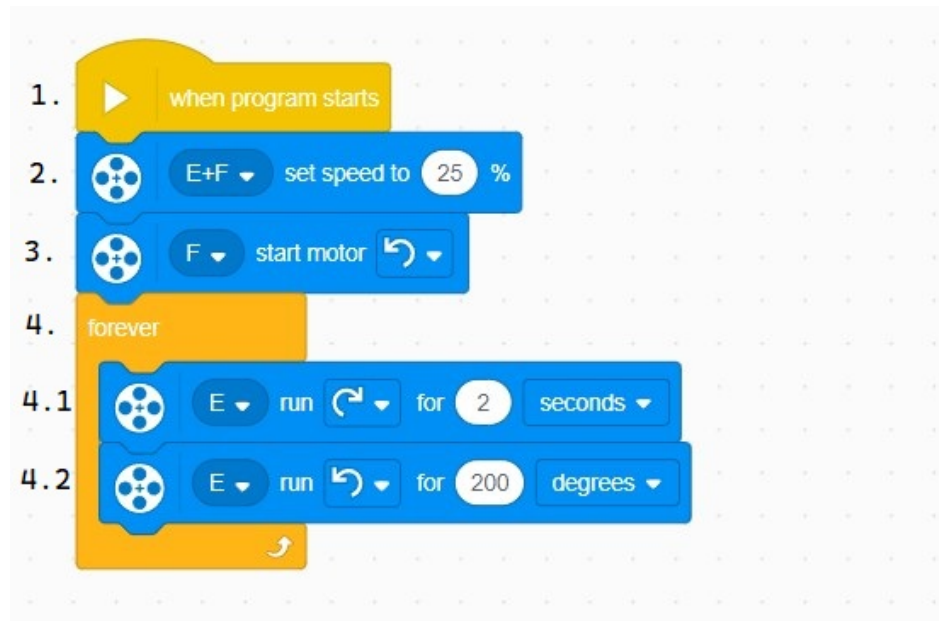
1.3 Vzorové programy

Teraz si ukážeme nejaké vzorové programy, ktoré vieme zostrojiť a spustiť v robotovy zostrojenom z tejto stavebnice. Tieto úlohy nájdeme v príručke [1].

1.3.1 Pohyb do štvorca

Počas celej tejto podkapitoli bude používať tohto robota (viď obr. 1.2)Na začiatku odštartujeme s jednoduchým programom pre robota, ktorý sa bude hýbať do štvorca. Po prejdení určitej vzdialenosti sa otočí na mieste a opäť ide vpred. Na obrázku 1.3 programu reprezentujúceho tento pohyb môžeme vidieť:

- 1. začiatok programu, ktorý sa spustí hneď po prijatí programu do riadiacej jednotky alebo po opätovnom stlačení ovladacieho tlačidla



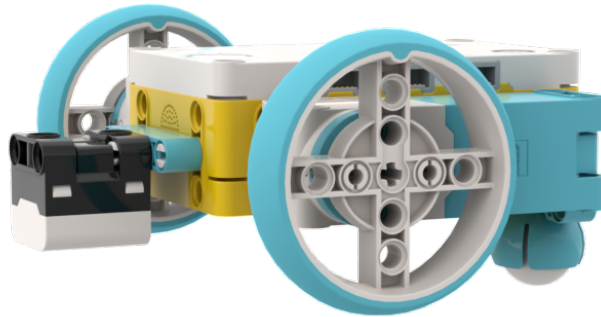
Obr. 1.3: Na obrázku je možné vidieť program pre robota, ktorý bude chodiť do štvorca.

- 2. nastavenie rýchlosti robota na 25% výkonu
- 3. spustenie motora, ktorý je pripojený na port F, v protismere hodinových ručičiek
- 4. cyklus programu, ktorý bude trvať pokiaľ robota nevypneme
- 4.1 spustenie motora, ktorý je na porte E, v smere hodinových ručičiek po dobu dvoch sekúnd
- 4.2 prepnutie tohto motora do opačnej strany pokiaľ neurobí pohyb veľkosti 200 stupňov (treba vziať do úvahy, že je to orientačná hodnota, ktorá závisí od povrchu koliesok, povrchu terenu, baterie riadiacej jednotky)

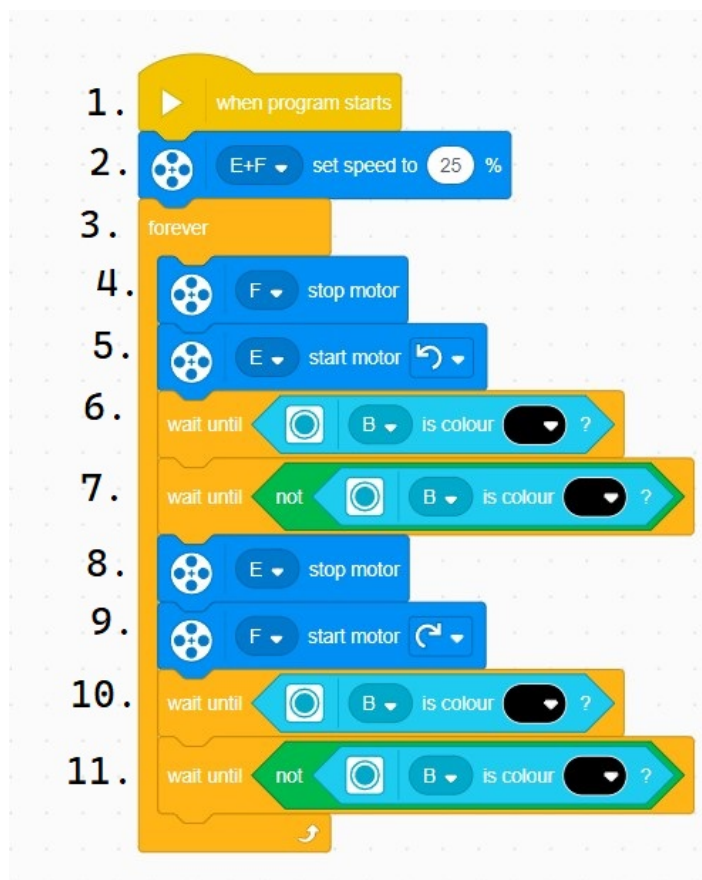
1.3.2 Sledovanie čiary

V tejto časti bude používať robota, ktorý je na obrázku 1.4, ktorý sa od predošlého líši iba pridaným senzorom na farby. Tentokrát bude robot nasledovať čiaru, na ktorú bol položený, inak sa bude krútiť dokolečka. Na obrázku 1.5 vidíme program zodpovedajúci popisu pohybu. Tu je jeho vysvetlenie:

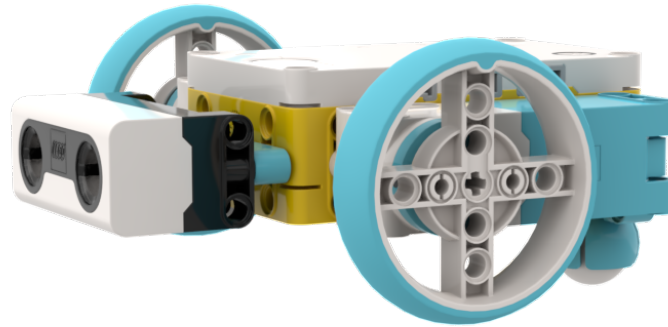
- 1. začiatok programu, ktorý sa spustí hneď po prijatí programu do riadiacej jednotky alebo po opätovnom stlačení ovladacieho tlačidla
- 2. nastavenie rýchlosti robota na 25% výkonu
- 3. nekonečný cyklus programu



Obr. 1.4: Na obrázku je možné vidieť robota, ktorého budeme používať v podkapitole 1.3.2. Obrázok bol vytvorený pomocou stud.io aplikácie.



Obr. 1.5: Na obrázku je program, vďaka ktorému robot sleduje čiaru, na ktorú je položený.



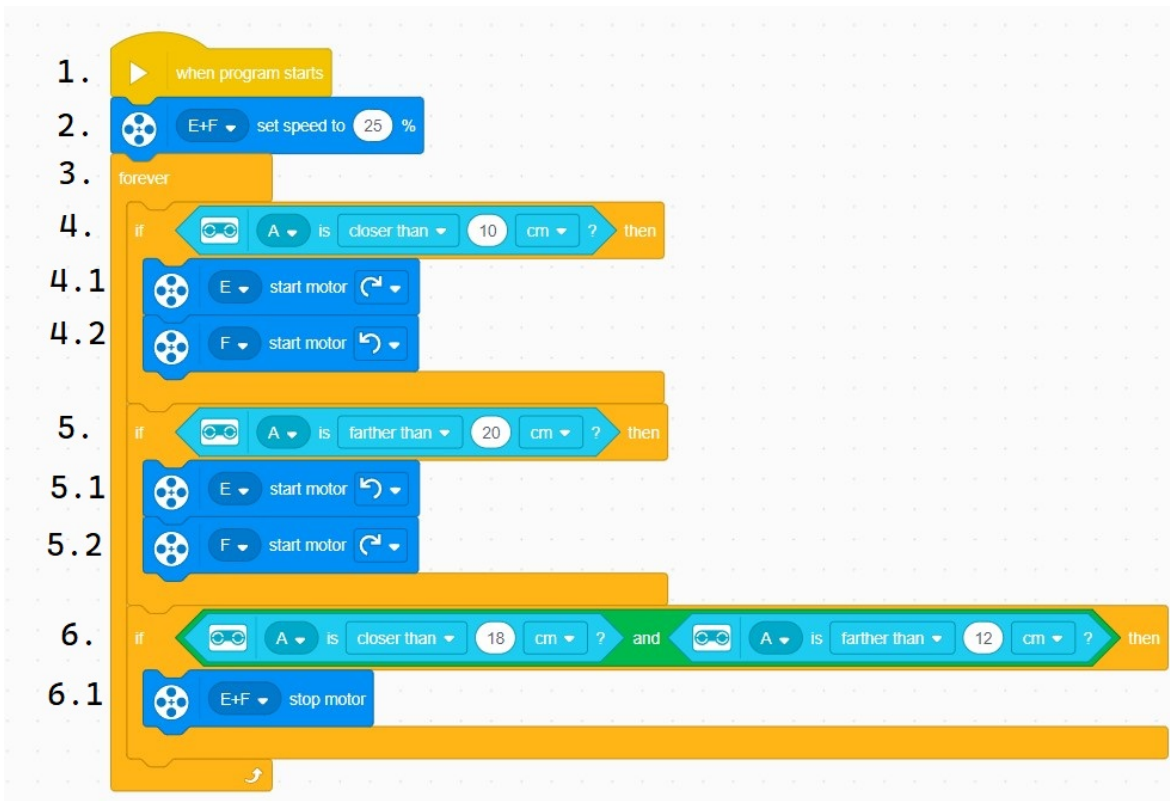
Obr. 1.6: Na obrázku je možné vidieť robota, ktorého budeme požívať v podkapitole 1.3.3. Obrázok bol vytvorený pomocou stud.io aplikácie.

- 4. zastavenie motora F
- 5. spustenie motora E v protismere hodinových ručičiek
- 6. tu program čaká, kým senzor nezasníma čiernú farbu
- 7. teraz program opäť čaká, kým nestratí čiernú farbu
- 8. zastavenie motora E
- 9. spustenie motora F v smere hodinových ručičiek
- body 10. a 11. zodpovedajú bodom 6. a 7.

1.3.3 Sledovanie objektu

Teraz budeme používať robota, ktorý je na obrázku 1.6, ktorý sa od prvotného robota líši iba pridaným ultrazvukovým senzorom. Robot bude sledovať objekt, ktorý je pred ním. Je to zjednodušená verzia príkladu z príručky [1] úloha 16. My ale použijeme iba jeden senzor. Obrázok 1.7 obsahuje tento program a tu je jeho vysvetlenie:

- 1. začiatok programu, ktorý sa spustí hneď po prijatí programu do riadiacej jednotky alebo po opätovnom stlačení ovladacieho tlačidla
- 2. nastavenie rýchlosti robota na 25% výkonu
- 3. nekonečný cyklus programu



Obr. 1.7: Na obrázku je program, vďaka ktorému robot sleduje objekt, ktorý je pred ním.

- 4. podmienka, v ktorej sa pýtame, či objekt pred nami je bližšie ako 10 centimetrov. Ak je táto podmienka splnená, tak:
 - 4.1 a 4.2 zapnutie motoru E v smere hodinových ručičiek a motoru F v opačnom smere
- 5. podmienka, v ktorej sa pýtame, či objekt pred nami je od nás ďalej ako 20 centimetrov. Ak je táto podmienka splnená, tak:
 - 5.1 a 5.2 zapnutie motorov E a F v opačnom smere ako v bodoch 4.1 a 4.2
- 6. podmienka, v ktorej sa pýtame, či objekt pred nami je ďalej ako 12 centimetrov a zároveň bližšie ako 18 centimetrov. Ak je táto podmienka splnená, tak:
 - 6.1 vypneme oba motory

Kapitola 2

Špecifikácia programu

V tejto kapitole bude popis návrhu a špecifikácie programu, ktorý bude slúžiť na tvorbu stavových automatov pre stavebnice Spike-prime, presnejšie pre podkategóriu obsahujúcu Prime hub, ktorý je špecifikovaný v podkapitole 1.1.

2.1 Špecifikácia platformy a jazyka

Program bude dostupný pre distribúciu operačného systému windows. Písaný bude v jazyku C# ako formulárová aplikácia pre windows.

2.2 Komunikácia programu s robotom

Na komunikáciu bude použitý bluetooth pomocou ktorého budú prenesené dáta vo formáte JSON. Program tento JSON bude tvoriť a firmware v robotovy čítať a vykonávať pokyny v ňom zakódované.

2.3 Rozhranie programu

Celé rozhranie programu bude v anglickom jazyku. Užívateľ bude nútení použiť pre vytváranie programu príkazy v anglickom jazyku avšak na pomenovanie stavov, premenných a funkcií bude môcť použiť akýkoľvek jazyk.

2.4 Stavový automat

Užívateľ bude môcť vytvoriť jednoduchý stavový automat alebo niekoľko stavových automatov, ktoré budú vykonávané paralelne. V prípade že automaty budú spustené paralelne, užívateľ bude musieť predom určiť, ktorý automat bude ovládať ktoré porty pre vykonávanie akcií, avšak porty na snímanie okolného sveta budú mať spoločné.

2.5 Stavy

K dispozícii budú tri typy stavov. Prvý a základný bude prázdny. Užívateľ ho môže iba pomenovať. Druhý je aktívny stav, do ktorého bude môcť užívateľ vložiť časť kódu a interval opakovania. Tento kód sa bude vykonávať pokiaľ sa program neposunie do ďalšieho stavu. Tretí stav, ktorý bude môcť užívateľ použiť je na tvorbu vnoreného stavového automatu. Po príchode programu sem sa program vnorí a začne vykonávať príkazy v danom automate.

2.6 Prechody medzi stavmi

Ku každému stavu bude možné pridať prechod do iného stavu. Tento prechod bude obsahovať podmienku za akej sa stav zmení a môže obsahovať zoznam príkazov, ktoré sa budú mať vykonať počas prechodu medzi stavmi. Podmienkou môže byť nasnímaná hodnota, hodnota premennej alebo aj vrátená hodnota stavového automatu, ktorý bol vykonávaný.

2.7 Inicializačný stav

Každý stavový automat bude mať na začiatku inicializačný stav, v ktorom si bude môcť užívateľ nastaviť premenné, priradiť porty alebo kopírovať a meniť globálne premenné.

2.8 Konečný stav

Stavové automaty budú mať aj konečný stav, ktorý sa vykoná pred opustením stavového automatu. Do tohto stavu sa bude môcť dostať prechodovým stavom alebo takzvaným superprechodom.

2.9 Superprechod

Je to stavový prechod, ktorý si bude môcť užívateľ vytvoriť. V ktoromkoľvek stave ak nastane podmienka uvedená v superprechode presunie program do konečného stavu. Superprechod bude môcť tiež obsahovať príkazy, ktoré bude program vykonávať pred tým, než prejde do konečného stavu.

2.10 Premenné

Užívateľ bude mať možnosť vytvoriť si lokálne premenné viditeľné iba pre daný automat alebo globalne premenné viditeľné pre celý program. Lokálne premenné sa po opustení daného stavového automatu vymažu.

2.11 Knižnica

Program bude umožňovať užívateľovi uložiť stavový automat, prechod alebo zoznam príkazov a znovu ich použiť neskôr. Táto knižnica sa bude inicializovať zo súboru a tak bude jednoduché pre užívateľov si tieto automaty, prechody a príkazy vymieňať.

Záver

...

Literatúra

- [1] Pavel Petrovič. Stavebnice lego mindstorms education ev3, materiály ku školeniam. *robotika.sk/events/18Skolenia/priruckaEV3.pdf*, pages 40–72, ????

Príloha A: obsah elektronickej prílohy

...

Príloha B: Používateľská príručka

...