

10. PARKOVACÍ ASISTENT

METODICKÉ POKYNY PRO UČITELE



K čemu aktivita slouží?

Žáci se v této aktivitě:

- naučí analyzovat a vysvětlit existující program,
- naučí upravit stávající program,
- vytvoří pokročilejší program s využitím ultrazvukového senzoru,
- naučí vytvářet vlastní programové bloky.



Přepokládaný rozsah aktivity

Minimálně 2x45 minut



Jak při řešení aktivity postupovat?

Úloha navazuje na předchozí práci s ultrazvukovým senzorem, takže by již nemělo být nutné žákům práci s ním na začátku hodiny blíže připomínat. Měli by být schopni sami začít řešit dílčí aktivity a otázky. Kontrolujte jejich práci, odpovídejte na případné dotazy a korigujte činnost žáků, aby došli ke správnému řešení.



Co si pro řešení aktivity připravit?

Pro testování schopnosti robota najít místo k zaparkování je třeba připravit vhodné místo. K tomu velmi dobře poslouží dvě nebo více krabic od stavebnice EV3. Můžeme je totiž různě posouvat, měnit velikost parkovacího místa a testovat, jestli je robot správně vyhodnotí jako dostačující, či příliš malé.



Popis částí aktivity, předpokládané řešení

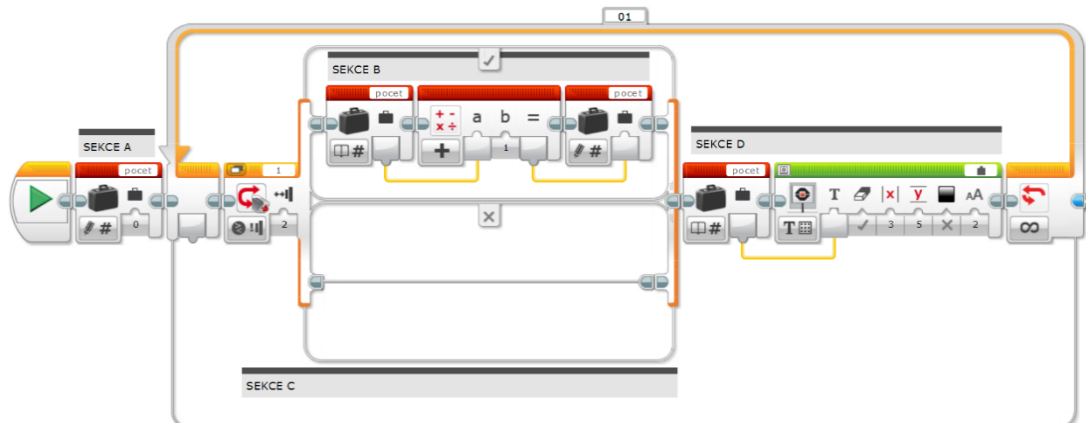


10.1 Dříve než začneme

Sekce je zaměřená na analýzu programu s využitím proměnné.



10.1.1 Poznáváme



Obrázek 1 – Program analyzovaný v záložce 10.1.1.

Zadání: Vysvětlete, k čemu slouží následující program a jeho sekce.

Sekce A: Vytvoření proměnné *pocet* a její inicializace (hodnota 0 se uloží do proměnné).

Sekce B: K hodnotě uložené v proměnné *pocet* se přičte hodnota 1 a výsledek se znovu uloží do proměnné *pocet*.

Sekce C: Blok podmíněného vykonání, který zajišťuje, že v případě stisknutí a uvolnění dotykového tlačítka (režim *Bumped*) je proveden úkon popsany v sekci B. V opačném případě se neprovede nic.

Sekce D: Hodnota uložená v proměnné *pocet* se po přičtení hodnoty 1 vypíše na displej řídicí jednotky.

Program slouží k: Počítá, kolikrát je stisknuto tlačítko dotykového senzoru, a tuto hodnotu vypisuje na displej řídicí jednotky.



10.1.2 Úkol

Otázka: Co by program po spuštění zobrazoval, pokud bychom v sekci A zadali místo 0 hodnotu 10?

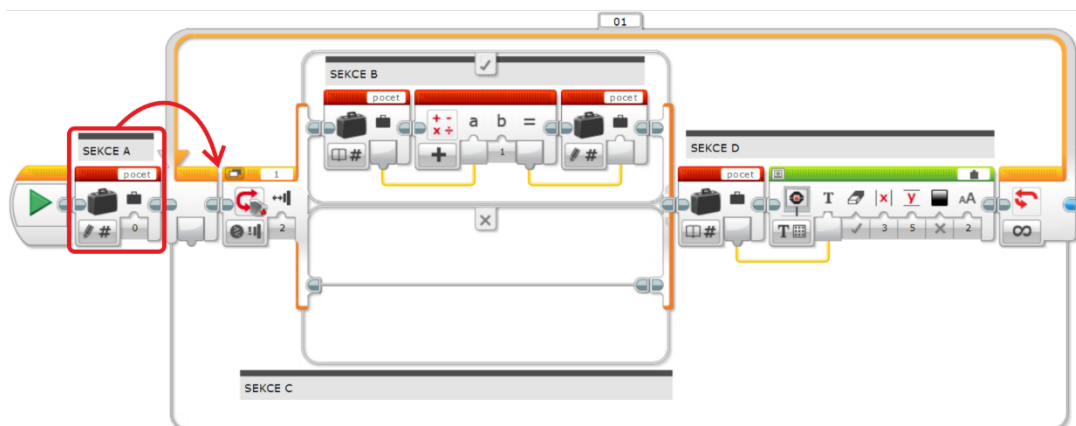
Odpověď: Funkce programu by byla stejná, ale přičítal by čísla tak, že by začínal od čísla 10.



10.1.3 Přesun bloku

Otázka: Vysvětlete, jak by program fungoval, pokud bychom blok ze sekce A vložili přímo před blok podmínky **Switch** (sekce C)?

Odpověď: Po stisku tlačítka by se hodnota navýšila o 1 a vypsala na displej. Toto vypsání bychom ale ani nepostřehli a hodnota v proměnné by se dalším průchodem cyklu znovu nastavila na 0 a program by čekal na další stisk tlačítka.



Obrázek 2 – Zadání úkolu 10.1.3.

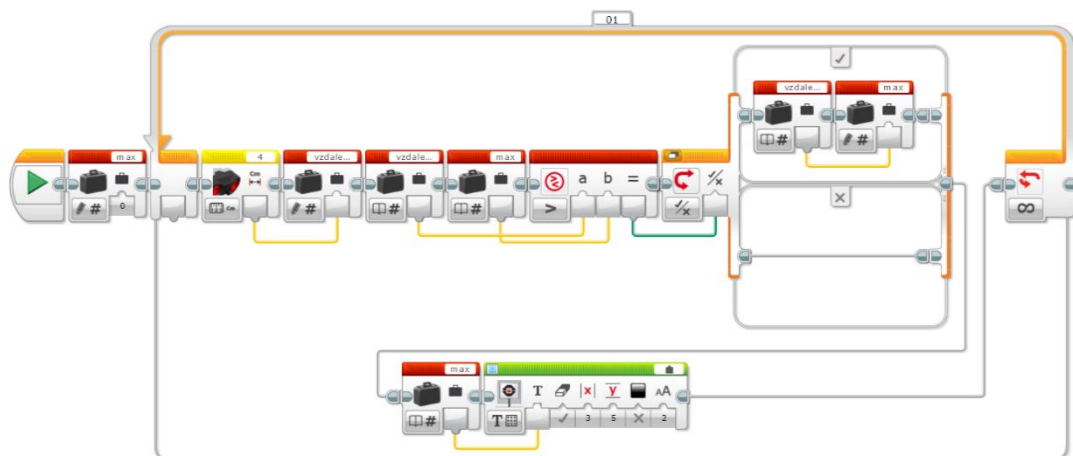
Otázka: Co musíme upravit, aby program počítal každý stisk za 10?

Odpověď: V sekci B musíme při stisku tlačítka přičítat hodnotu 10 místo 1.



10.1.4 Proměnná

Zadání: Vysvětlete, jakou funkci má program na obrázku.



Obrázek 3 – Zadání úkolu 10.1.4.

Odpověď: V programu je na začátku vytvořena proměnná *max*, jejíž hodnota je nastavena na 0. Po celou dobu chodu programu měří ultrazvukový senzor vzdálenost, kterou ukládá do proměnné *vzdálenost*. Tyto dvě proměnné jsou následně porovnávány, a pokud je změřená vzdálenost větší než maximum, uloží se tato hodnota do proměnné *max* a vypíše se na displej. Zjednodušeně tedy můžeme říct, že program provádí neustálé měření a na displej zobrazuje největší naměřenou hodnotu.

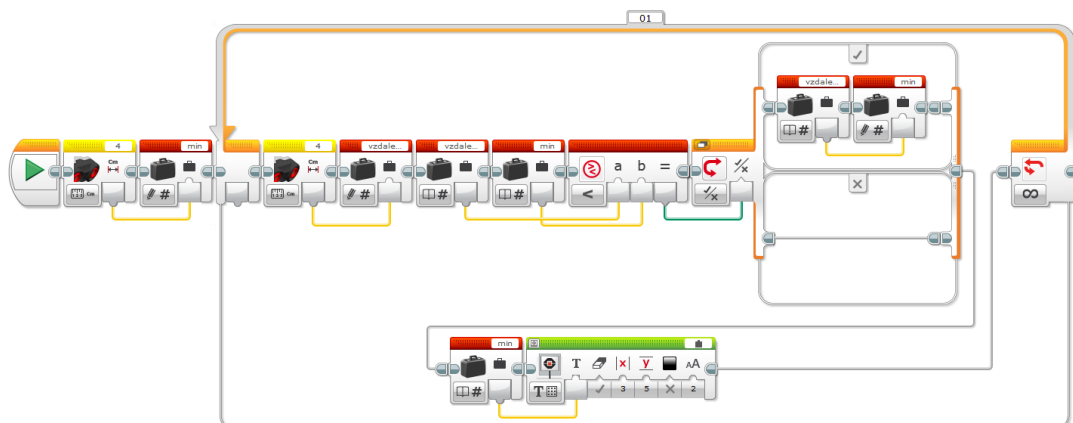


10.1.5 Úprava

Žáci pracují s programem z úkolu **10.1.3**, který si otevřou prostřednictvím odkazu.

Otázka: Co bychom museli změnit, aby program zjišťoval minimální vzdálenost?

Odpověď: V programu musíme provést dvě změny. Pro lepší názornost jsme ve správném řešení, znázorněném na obrázku 4, přejmenovali proměnou *max* na *min*. Na začátku musíme smazat nulování proměnné. Místo něj si musíme do proměnné *min* uložit první a tedy i aktuálně nejnížší hodnotu. Druhou změnu provedeme v bloku **Compare**. Hledáme nejnížší hodnotu, takže zjišťujeme, zda je naměřená vzdálenost menší než aktuální hodnota v proměnné *min*.



Obrázek 4 – Správné řešení úkolu 10.1.5.



10.2 Jak bude parkovací asistent fungovat?

Úvodní text seznamuje žáky s fungováním parkovacího asistenta. Důležitá je zde informace, který senzor žáci využijí. V minulé hodině umístili ultrazvukový senzor na čelní část, aby při jízdě snímal překážku před sebou. Nyní je třeba umístit ho na pravý bok, aby mohl snímat řadu podélně zaparkovaných vozidel. Ověřte tedy, že žáci umístili senzor správně. Předejdete tak pozdějším problémům.



10.2.1 Funkce

Zadání: Vymyslete, jak by náš robot mohl automatizovaně vyhledat volné místo v řadě aut a do tohoto místa bezpečně zaparkovat.

Společně se žáky diskutujte a poznamenávejte si (např. na tabuli) navržená řešení, jejich výhody a nevýhody.

V rámci diskuze nechte žáky přicházet s vlastním řešením. Podporujte je, aby přicházeli např. na to, jak může robot detekovat volné místo, jak pozná, že tam je, či není překážka, jak vyhodnocuje, zda je místo dostatečně veliké a co znamená, že robot do volného místa zaparkuje. Nezabíhejte však do podrobných technických řešení. Na ty bude prostor v dalších částech. Pokud žáci nebudou na otázky reagovat, nechte jim čas je promyslet.

Nevadí ani, pokud budou chtít žáci řešit v rámci diskuze i fáze rozhodování, které ale jinak probíráme v další samostatné záložce.

Možné otázky:

- Jak může robot poznat, kde začíná volné místo?
- Co bude dělat v případě, že ho najde, a co, když ho nenajde?
- Jak zjistíme, že je mezera dostatečně veliká pro zaparkování robota?
- Jak poznáme, zda na místě pro zaparkování není nějaká překážka?
- Jak by mohl robot zaparkovat?



10.2.2 Části

Otázky: Které fáze bude mít řešení našeho problému? Co bude robot řešit?

Na tomto místě se žákům snažíme vysvětlit, že řešení problému je poměrně náročné a rozsáhlé. Bude tedy třeba si ho rozdělit na menší, snáze řešitelné části. Poznámky žáků můžete pouze zkontrolovat nebo je společně diskutovat.

Řešení by se mělo skládat z několika částí:

- nalezení začátku volného místa,
- změření vzdálenosti potřebné pro zaparkování,
- zaparkování,
- detekce případné překážky znemožňující zaparkování (rozšiřující aktivita).



SOS

10.2.3 SOS části

Nápověda, ve které mohou žáci najít odpovědi na otázku 10.2.2. Přistupovat by k ní ale měli pouze v případě, že si s odpovědí nevědí rady.



10.3 Nalezení začátku volného místa



10.3.1 Aktivita

Zadání: Vytvořte program, díky kterému robot detekuje začátek volného místa pro zaparkování. Po nalezení místa zastaví a oznámí to dlouhým výrazným tónem.

Řešení: K řešení aktivity je zapotřebí využít blok **Wait** (čekání), řízený hodnotou zjištěnou ultrazvukovým senzorem. Tento konstrukt by již měli žáci velmi dobře znát z řešení úkolů v předchozích kapitolách. Možné správné řešení vidíte na obrázku 5. Robot jede neustále vpřed, a jakmile se prodlouží naměřená vzdálenost (objeví se mezera mezi zaparkovanými vozidly), zastaví a vydá výstražný tón.



Obrázek 5 – Detekce začátku volného místa na zaparkování.

Otázka: Jak senzory robota poznají, že našly volné místo pro zaparkování?

SOS

10.3.2 Nápověda

Pokud by si žáci s řešením aktivity nevěděli rady, mají k dispozici krátkou nápovědu. Ta by je měla navést na to, že musí detekovat větší naměřenou vzdálenost než tu, kterou vrací senzor v případě, když na podélném parkovacím místě stojí automobil.



10.3.3 Řešení

Otázka: Může být tento program správným řešením aktivity? Pokud ne, proč?

Odpověď: Jestliže ani po nápovědě žáci nevědí, jak úlohu řešit, je pro ně připravena záložka **Řešení**. Zde je na obrázku náš výsledný program, který ovšem obsahuje chybu. Pokud žáci chybu na první pohled nevidí, měli by program otestovat a zjistit, jak se chová. V programu je otočený operand. Začátek volného parkovacího místa je vyhodnocen v případě, že je vzdálenost menší než 25 cm. Žáci by měli tuto chybu najít a opravit (viz obrázek 6).



Obrázek 6 – Program pro detekci začátku volného parkovacího místa s vyznačenou chybou.



i

10.4. Tvorba vlastního bloku

Další část je teoretická a žáci se v ní naučí, jak si vytvořit vlastní blok. Jelikož program pro vytváření parkovacího asistenta bude poměrně rozsáhlý, může se jim tato funkce hodit. Seznámí se zde s průvodcem pro vytváření nového bloku. Následně jsou vyzváni, aby si z předchozího programu zkusili vytvořit vlastní blok.



Upozornění: Důležité je, aby si při vytváření vlastního bloku správně označili všechny bloky, které budou jeho součástí. Nesmí ovšem vybrat úvodní blok pro spuštění programu (**Start**). Tuto skutečnost žákům raději zdůrazněte.



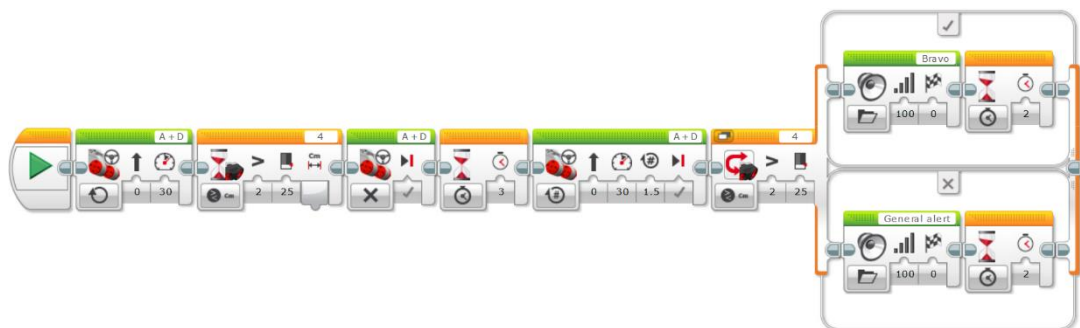
10.5 Měření potřebné vzdálenosti



10.5.1 Aktivita

Zadání: Vytvořte program, pomocí kterého robot změří, zda je místo pro zaparkování dostatečně velké, a pokud ano, bude nás o tom informovat. Pokud ne, vydá chybové hlášení nebo tón.

Řešení: V tomto bloku žáci rozšíří již vytvořený program. Po detekci začátku volného místa musí nyní zjistit, zda je parkovací místo dostatečně velké. Nejjednodušší způsob je, že po detekci začátku volného místa popojede robot o vzdálenost, která odpovídá délce robota. Zde znovu ověří, jestli je zde volno. Možné řešení aktivity vidíte na obrázku 7. Pokud je místo dostatečně velké, ozve se tón *Bravo*. V opačném případě zazní chybové hlášení *General Alert*.



Obrázek 7 – Možné řešení aktivity pro změření potřebné vzdálenosti.

SOS

10.5.2 SOS Náповěda

Sekce nápověda obsahuje vodítko k tomu, jak aktivitu vyřešit. Na obrázku v učebnici je vidět, co může nastat, pokud je na místě, kde robot měří vzdálenost, zaparkované auto.



10.5.3 Otázka

Závěrečné otázky jsou pouze pro zamyšlení. Žáci mají za úkol zmapovat možné problémy, které by při realizaci úlohy tímto způsobem mohly nastat. Jedná se hlavně o to, že v místě, které si robot odměří, může být například zaparkován motocykl, umístěna popelnice nebo lampa veřejného osvětlení. To by robot nedetekoval, protože neověřuje volnost místa i během popojíždění. Došlo by tak ke kolizi. Na řešení této situace je zaměřena následující rozšiřující aktivita.



10.6 Rozšiřující aktivita pro rychlé žáky



Úvod k rozšiřující aktivitě je informativní. Pokud důvody nepraktičnosti řešení nenalezli žáci sami, zde je mají krátce nastíněny. Zároveň zjistí, co bude jejich dalším úkolem.



10.6.1 Aktivita

Zadání: Analyzujte následující program a zjistěte, k čemu slouží.

Řešení: Řešení této aktivity je poměrně náročné. Proto ji uvádíme jako rozšiřující, pro rychlé žáky. Úvodní aktivita v tomto bloku se zaměřuje na analýzu programu. V konstrukturu na obrázku je použit blok pro měření počtu otáček (**Motor Rotation**). Počet naměřených otáček je v úvodu vynulován. Pokud je hodnota zjištěná senzorem menší nebo rovna 25 cm, je logická proměnná *obsazeno* nastavena na **True** (pravda). Pokud během ujetí potřebných otáček značí proměnná *obsazeno*, že místo není volné k zaparkování, je to pro robota signál, že zde zaparkovat nemůže. Konstrukturu tedy ověřuje, zda je parkovací místo volné po celé délce, tzn. že měří vzdálenost i během popojíždění.



10.6.2 Náповěda

Náповěda v další záložce obsahuje vysvětlení a představení bloku **Motor Rotation**. Její přečtení může žákům usnadnit analýzu programu v sekci 10.6.1 Aktivita.



10.6.3 Návod

Na závěr je žákům vysvětleno, jak by měl celý program fungovat. Které části vytvořené průběžně by do něj měli zahrnout a co by měly zajišťovat. Realizace se samozřejmě může v různých skupinách drobně lišit.



10.7 Parkujeme

V této sekci je realizována poslední část funkce parkovacího asistenta, kterou je zaparkování v řadě vozidel.



10.7.1 Program

Zadání: Vytvořte program, pomocí kterého robot zaparkuje do volného parkovacího místa.

Řešení: Žáci by již měli znát, jak správně ovládat motory robota a měnit jejich směr. Musí si zde ovšem uvědomit, že robot bude do řady vozidel couvat. Musí tedy používat zápornou hodnotu výkonu motorů (např. -50 %).



10.7.2 SOS Nápověda

Drobná nápověda a připomínka toho, že s pohybem se už žáci setkali v jedné z předchozích kapitol.



10.7.3 Otázky

Záložka slouží jako podnět k závěrečné diskuzi. Žáci zde mohou společně zhodnotit svá řešení, jejich výhody a nevýhody a také diskutovat, zda by tento princip mohl obstát i v silničním provozu v reálném světě.

Otázky:

Mohl by takovýto princip parkování fungovat v reálném životě?
Které problémové situace by mohly nastat?