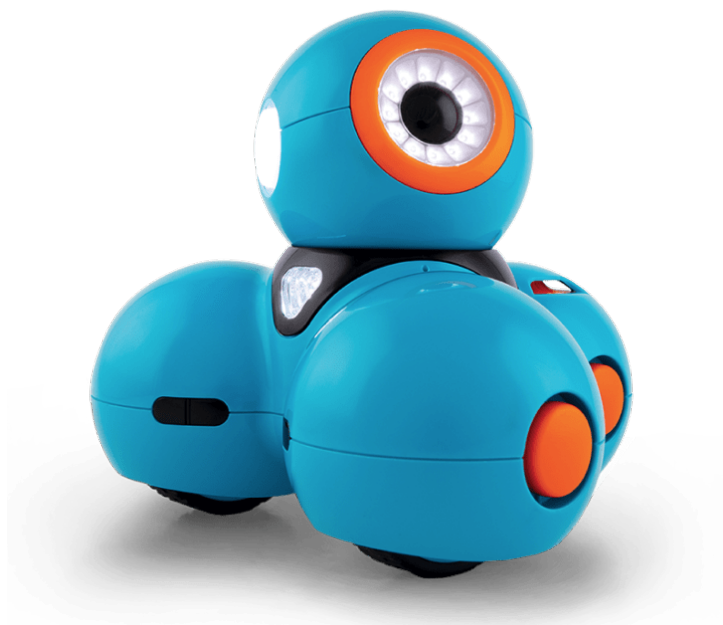


# Rozvíjení algoritmického myšlení pomocí robotů Dash a Dot



## 6.1 Pohybové úlohy

Během práce na pohybových úlohách se žáci postupně seznámí s nejrůznějšími dovednostmi v oboru programování. 1. úloha funguje jako návod pro samotné připojení robotů a prostředí blockly. Žáci začnou na jednoduchých úlohách, kde budou využívat hlavně robota Dashe a jeho pohyb. Postupně se k pohybovým blokům přidá využívání cyklů, podmínek, a nakonec v rozšířeních u některých úloh i využití proměnných. Ve stejné posloupnosti, jak jsou úlohy v práci číslované by měly být i následně využívány. Úlohy postupují podle obtížnosti a postupně se nabalují nové vědomosti. Všechny pohybové úlohy se zaměřují hlavně na Dashe a jeho pohybové dovednosti. Dot je zde využíván minimálně.

### 6.1.1 Začínáme programovat

#### Zadání:

Nejdříve začneme tím, že pomocí podržení tlačítka oba roboty zapneš. Následně můžeš otevřít aplikaci blockly, kde pomocí oranžového plus v pravém horním rohu připojíš oba roboty. Roboti se k zařízení připojují pomocí Bluetooth, tak si dej pozor abys ho měl zapnuté a povolené. Úspěšné připojení poznáš tak, že ikony obou robotů zezelenají stejně jako na obrázku.



Další důležitý krok, než začneme programovat je vytvoření nového projektu. Pomocí Menu -> My Projects -> Create New -> Blank Project si vytvoříš nový projekt. Dbej zvýšenou pozornost tomu, jak projekt pojmenuješ, ať v budoucnu nemáš problém mezi sebou projekty rozeznat. Pokud si vše udělal správně, měl by si na ploše vidět pouze blok When Start.

#### Cíle a poznámky k úloze:

Co se týče úlohy Začínáme programovat tak jejím cílem není cílem dostat od žáků nějaký jimi vytvořený projekt, ale jedná se spíše o prvotní seznámení s roboty a s prostředím ve kterém budou pracovat. Měla by být doplněna o lehký výklad k samotnému blokovému prostředí kde by měly být popsány jeho nejdůležitější bloky a funkce. V případě, že učitel ví o věcech, které jeho žákům dělají problémy není od věci jim tyto problematiky znovu dovysvětlit, aby se předešlo pozdějším žakovským problémům s ostatními úlohami.

## 6.1.2 První krůčky

### Zadání:

Pokud se ti povedlo úspěšně dokončit připojení robotů a vytvořit si svůj vlastní projekt, můžeme začít s 1. úlohou. Budeme se věnovat základnímu pohybu, který může připadat jednoduchý, ale jedná se o základní stavební kámen pro všechny ostatní úlohy.

V této úloze bude tvým úkolem vyzkoušet několik bloků z kategorie Drive, konkrétně se bude jednat o bloky Forward, Backward, Turn Left a Turn Right a následně tyto bloky využít k vytvoření Dashova pohybu. Postupně využívej všechny bloky a pečlivě sleduj, jak se Dash chová. Jistě sis všiml, že jsou zde defaultně nastaveny nějaké hodnoty. Co když tyto hodnoty změníš? Jak to ovlivní chování Dashe?

Nakresli pomocí robota písmeno L.

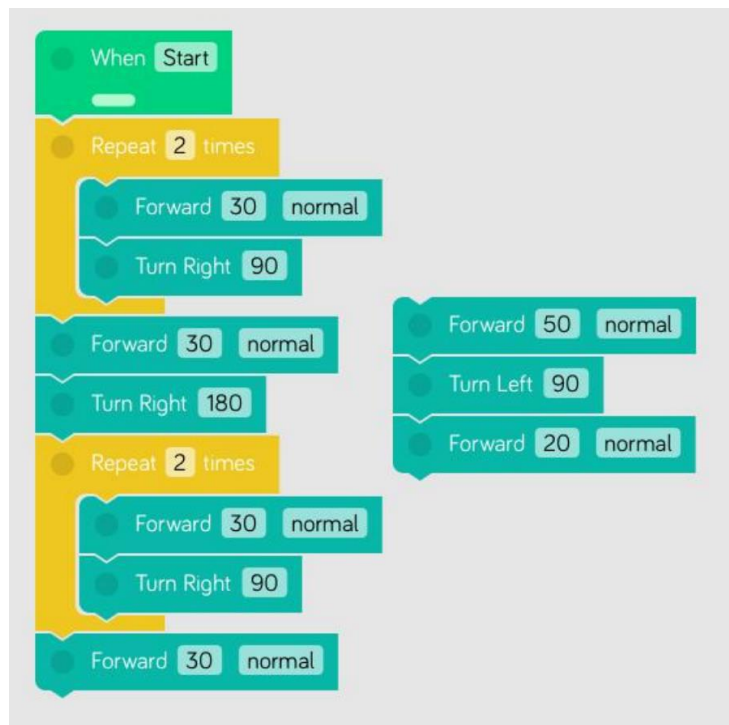
**Rozšíření:** Vytvoř program, díky kterému Dash nakreslí číslo 3. Číslo může být hranaté, můžeš se inspirovat například sedmi-segmentovým displejem, který se využívá například u digitálních hodin. Robot může začínat na kterémkoli konci.

### Cíle a poznámky k úloze:

V úloze s názvem První krůčky se žáci seznámí s konkrétními pohybovými bloky a jejich výstupem bude 1. smysluplný program. Žáci si osvojí pohybové bloky a pokusí se robotem nakreslit číslo L. Zadání není nijak více specifikováno, takže hodnotit musíme hlavně to, jak žáci k problému přistoupili a zda po spuštění programu robot udělá požadovaný pohyb. Pomocí doprovodných otázek v zadání se v žácích snažím vzbudit větší zájem a to, aby se o svém programu pokusili více přemýšlet, a ne pouze splnit zadání.

Co se týče rozšíření, tak zde již žáci pouze aplikují vědomosti získané díky pozorování robotů. U této úlohy není důležité využití cyklů, proto není problém, pokud žáci vytvoří dlouhý program bez opakování. Nejdůležitější je, aby si dostatečně osvojili bloky ovládající pohyb robota, protože se jedná o základní stavební kámen pro všechny ostatní úlohy. Je pouze na nich, zda se rozhodnou využít například couvání, nebo robota otočí o 180 stupňů.

### Vzorové řešení:



U této úlohy neexistuje jediné správné řešení. Tím, že v úloze není definováno, zda má Dash například couvat nebo podobně mohou žáci využít téměř libovolný způsob. Musí ale samozřejmě dosáhnout požadovaného výsledku. Kontrolujeme tedy hlavně samotný pohyb robota, zda vytvořený pohyb opravdu připomíná písmeno L případně číslo 3. V zadání není nikde definován rozměr požadovaných útvarů, takže velikost je v tomto případě čistě na žácích.

### 6.1.3 Čtverec

#### Zadání:

Super, základní pohyb máme za sebou, a tak můžeme začít tvořit trochu komplexnější programy.

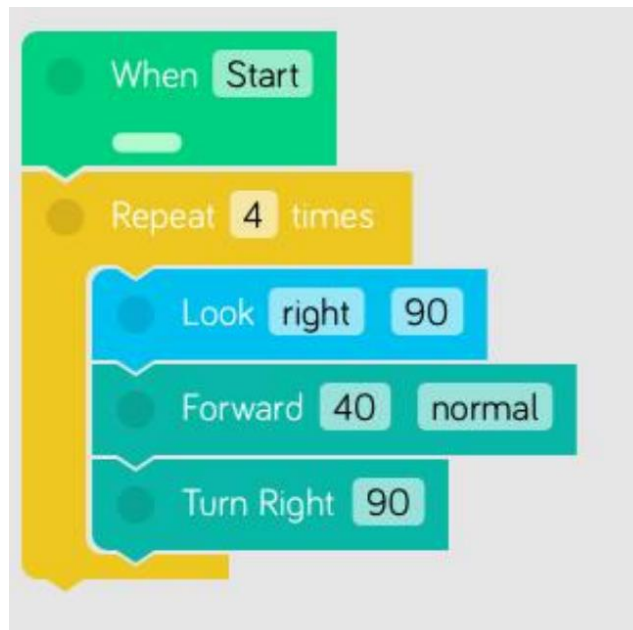
V této úloze bude tvým úkolem vytvořit program, díky kterému Dash udělá pohyb do čtverce. Strana čtverce by měla být dlouhá 40 cm. Dej si pozor na to, aby Dash skončil ve stejném bodě jako začínal. Jestli již Dash umí čtverec, pomocí dalších bloků zajisti, aby měl hlavu při jízdě vždy otočenou do středu čtverce.

**Rozšíření:** Pokud se ti v programu bloky hodně opakují, nešlo by program zkrátit? Vylepši program tak, aby byl každý blok vždy použit maximálně jednou. K tomu by ti mohly být užitečné některé bloky z kategorie Control.

#### Cíle a poznámky k úloze:

Úloha Čtverec přímo navazuje na předchozí úlohu a jejím hlavním cílem je procvičit doposud získané dovednosti ohledně pohybu a navázat na ni novou dovedností v podobě použití cyklů. Žáci mají za úkol vytvořit program díky kterému se robot na podlaze bude pohybovat do čtverce a poté přidat několik nových bloků pro náklon jeho hlavy. Následně jsou v rozšíření úlohy vybízeni k tomu, aby využili nové bloky a jimi vytvořený program tím hodně zkrátili. V této úloze je již velmi důležité, aby učitel při kontrole dal větší pozornost tomu, zda žáci opravdu cykly použili. Úloha u studentů prohlubuje schopnost analyzovat problém a strategicky přemýšlet o tom, jak nejlépe dosáhnout požadovaného řešení.

### Vzorové řešení:



U této úlohy je již důležité dbát na to, aby žáci správně využili cykly. Jako správné řešení tedy nemůžeme považovat příliš se opakující bloky i přesto, že se Dash bude pohybovat správně. Také je zde přidáno otáčení hlavy. Pokud budeme kontrolovat správnost řešení, musíme tedy dbát na to, zda žáci splnili opravdu všechny části zadání.

#### **6.1.4 Pozor!**

##### **Zadání:**

Jak název napovídá, podstatou této úlohy bude zjišťování, zda nemá Dash kolem sebe nějakou překážku, do které by mohl nabourat.

Tvým úkolem bude Dashe rozpohybovat tak, aby se rozjel rovně dopředu. V případě, že před sebou uvidí nějakou překážku, tak se zastaví a následně začne couvat. Pokud během couvání Dash detekuje překážku za sebou, opět se zastaví a pojedje vpřed.

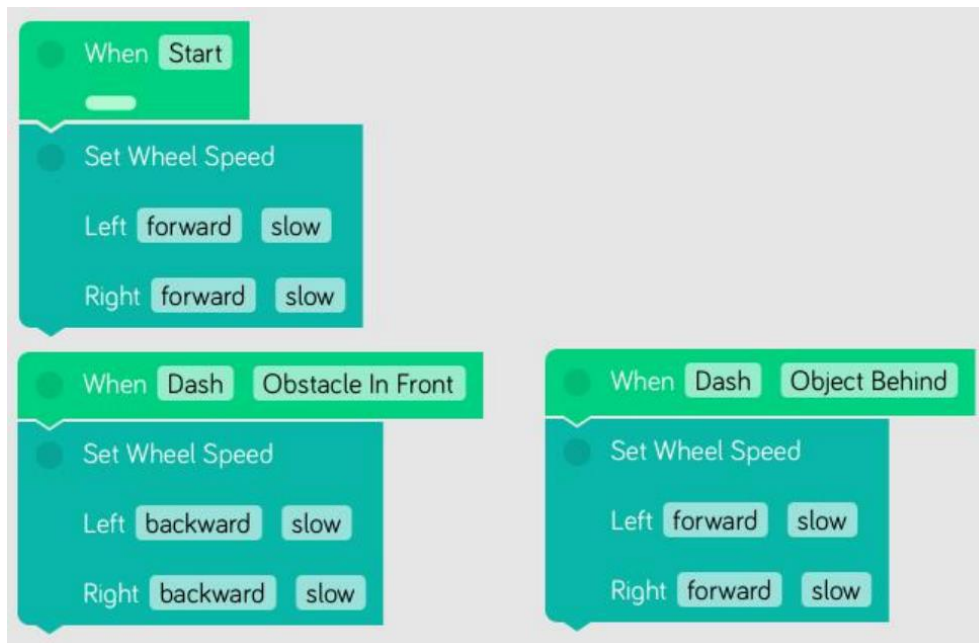
Nastav některou z pomalejších rychlostí pohybu, aby se dalo fungování programu pořádně vyzkoušet. Dej si pozor na to, aby byl program po spuštění funkční až do chvíle než jej někdo přeruší nebo vypne.

K vyzkoušení, zda program správně funguje můžeš využít například sešit ale klidně i Dashova pomocníka Dota.

##### **Cíle a poznámky k úloze:**

Hlavním cílem úlohy s názvem Pozor! je žáky seznámit se skutečností, že roboti disponují svými senzory a naučit je s nimi pracovat. Žáci budou tvořit takový program, aby Dash během svého pohybu monitoroval okolí a vždy před překážkou začal buďto couvat nebo naopak jel dopředu. Bez splnění této úlohy budou mít žáci velký problém s ostatními úlohami v tomto bloku, protože nebudou mít procvičenou práci se senzory. Jedná se tedy o prvotní seznámení se s čidly, které žákům usnadní práci v následujících úlohách. Vzorové řešení:

### Vzorové řešení:



U této úlohy by žáci s řešením neměli mít nějaký velký problém. Jedná se sice pro první využití senzorů, ale pokud využijí správný blok, samotné sestavení do funkčního celku by již mělo jít bez problému. Tato úloha by se samozřejmě dala splnit i pomocí podmínek if. Využil jsem ale raději bloku When, místo použití podmínky if, protože se robot v reálném prostředí a čase choval přirozeněji a přesněji.

### 6.1.5 Hra na schovávanou

#### Zadání:

V této úloze navážeme na znalosti a dovednosti získané v předchozí úloze a využijeme dota jako hledanou osobu.

Nejdříve si vymeze prostor, ve kterém budou oba roboti hru hrát. Mělo by se jednat o obdélník nebo čtverec. Prostor si označ nějakými předměty, a hlavně si změř délky stran, abys s nimi mohl v programu pracovat. Samotná hra bude probíhat tak, že kdekoliv ve vymezeném prostoru postavíš Dota. Dash postavíš do pravého dolního rohu vymezeného prostoru a vytvoříš program, díky kterému Dash Dota v prostoru vždy najde. Ve chvíli, kdy se k němu přiblíží tak pomocí zvukového znamení vyjádří, že ho našel. Mysli na to, že v prostoru se smí nacházet pouze Dot. Dash nesmí nikdy vyjet ven z označeného prostoru ani narazit do dota.

**Rozšíření:** Pokud již máš řešení hotové, vyzkoušej, zda opravdu funguje pro každý prostor, který tě napadne. Nešlo by pomocí jedné hodnoty určit velikost prostoru? Abys nemusel vždy při změně přepisovat celý program? Zajisti, aby program šel změnit pro jiné rozměry přepsáním pouze jedné hodnoty.

**Tip:** Prostor si nejdříve nakresli na papír a zamysli se, jak by to bylo možné udělat. K tomu, abys vždy při změně prostoru musel přepsat pouze jednu hodnotu ti pomůže proměnná, kterou najdeš v kategorii Variables.

#### Cíle a poznámky k úloze:

Úloha s názvem Hra na schovávanou navazuje na již získané dovednosti. Konkrétně rozvíjí zkušenosti s pohybem Dashe, použití cyklů, zjištění, zda se někde nenachází překážka a k tomuto přidává díky svému rozšíření novou znalost, a to v podobě proměnné. Hlavním cílem je rozšířit u žáků znalosti ohledně cyklů tím, že jsou nuceni použít jiný cyklus než pouze s konkrétním počtem opakování. Dále se úloha zaměřuje na rozvoj prostorové orientace a schopnost rozvrhnout si práci před samotným výkonem. Úloha vyžaduje vytvoření programu pro navigaci robota v předem určeném prostoru tak, aby pokryl všechny jeho části a našel svého parťáka Dota. Žáci dále získávají praktické zkušenosti v oblasti robotického programování a logického myšlení. V rozšíření se znovu pomocí otázek snažím v žácích vytvořit zájem a podněcovat je k tomu, aby o programu přemýšleli více obecně a ne pouze 37 tak, aby splnili zadání. Tímto žáky motivuji k využití proměnné, aby bylo zadání funkční i v případě změny rozměrů prostoru. Důležité je také to, aby si žáci uvědomili, že jediný rozměr, který potřebují k úspěšnému zvládnutí je délka spodní strany jimi vybraného tvaru, protože Dash se bude pohybovat pouze do chvíle, než najde Dota.

### Vzorové řešení:

```
When Start
Repeat until Dash Obstacle In Front
  Turn Left 90
  Forward 40 slow
  Turn Right 90
  Forward 10 slow
  Turn Right 90
  Forward 40 slow
  Turn Left 90
  Forward 10 slow
When Dash Obstacle In Front
  Stop Wheels
  Say Dash Ta Da!
```

Toto řešení je pro základní část úlohy. Jedná se pouze o to, aby žáci dokázali správně vymyslet systém, kterým se má robot po vytvořené oblasti pohybovat. Následně z této myšlenky vytvořit program, kde využijí již nabitě zkušenosti. Robota musíme ještě pomocí bloku When zastavit, jinak program dobíhá až do konce, čímž by se oba roboti mohli srazit.

```
When Start
Set [ ] = 40
Repeat until Dash Obstacle In Front
  Turn Left 90
  Do Drive [ ] cm
  Turn Right 90
  Forward 10 slow
  Turn Right 90
  Do Drive [ ] cm
  Turn Left 90
  Forward 10 slow
When Dash Obstacle In Front
  Stop Wheels
  Say Dash Ta Da!
```

Zde je složitější řešení, a to s použitím proměnné, díky které mohou žáci libovolně měnit rozměry daného útvaru. Kvůli své náročnosti by mohlo sloužit například jako zábava pro rychlejší žáky, zatímco ostatní budou pracovat na základní části této úlohy. Pokud víme, že pracujeme s šikovnými žáky, tak není problém rovnou od začátku vyžadovat rozšířené řešení

## 6.1.6 Slalom

### Zadání:

Jistě už si někdy při nějakém sportu běhal slalom, i kdyby ne tak nezoufej. V této úloze budeš mít příležitost si to vyzkoušet s robotem.

Vytvoř alespoň ze 4 stejně velkých předmětů pravidelný slalom. Použít můžeš například lahve od pití, důležité je, aby byly překážky přibližně stejně velké a pro robota dostatečně viditelné. Tvým hlavním úkolem bude vymyslet program, díky kterému Dash slalom projede, aniž by zapomněl na nějakou překážku nebo něco shodil. Využij minimálně 4 překážky. Vzdálenost mezi předměty může být stejná.

**Rozšíření:** Pokud již Dash slalom projede bez problémů, můžeme postoupit trochu dále. Vyndej ze slalomu některou z překážek, nebo mezi nimi například udělej různé mezery. Pokud Dash naráží, zajisti, aby slalom projel vždy, nehledě na to, jak vzdálené od sebe překážky budou

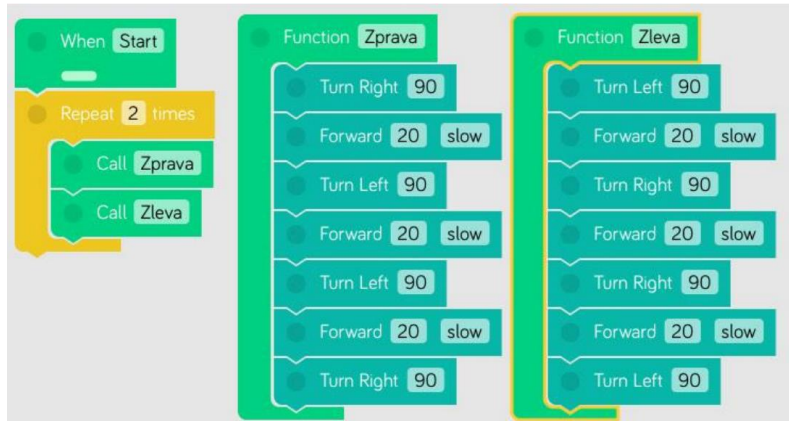
**Tip:** Nejdříve naprogramuj Dashův pohyb kolem překážky a ve finálním programu ho dále používej jako blok Call Function. Tímto se budeš v programu lépe orientovat. Podívej se tedy do kategorie Start, kde najdeš vše potřebné. K tomu, aby Dash střídal objíždění z levé a pravé strany využij proměnou.

### Cíle a poznámky k úloze:

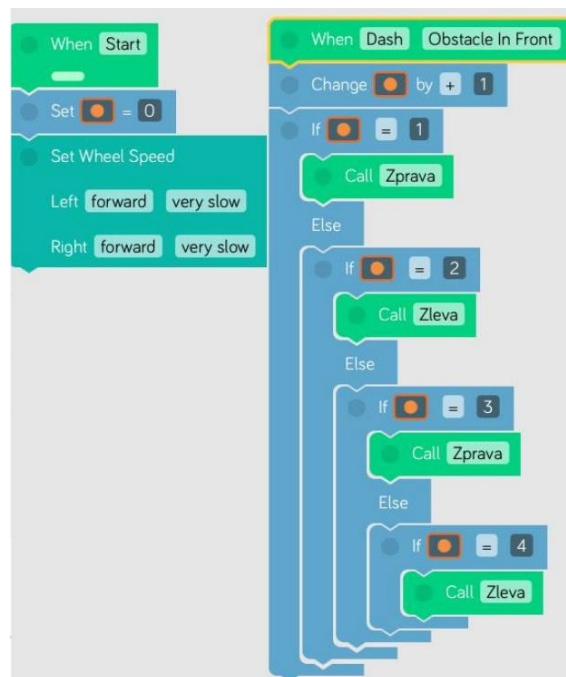
Slalom je poměrně prostorově náročná úloha čili není vhodná, pokud máte hodně žáků na ne moc velkém prostoru. Když tak je možnost využít místo této úlohy zadání z úloh 2 C a 2 D, které se zaměřuje na podobnou znalost nebo můžete pro toto řešení žáky rozdělit do větších skupin. Co se týče překážek tak já jsem pro vzorové řešení použil obyčejné ruličky od toaletního papíru a fungovaly bez problému. Žáci mají za úkol nejdříve vytvořit slalom mezi překážkami, které jsou od sebe stejně daleko. Hlavním cílem této úlohy je to, aby si žáci ozkoušeli, zda jimi ovládaný robot překážky objíždí, aniž by do některé narazil. Dbejte pozornost tomu, aby žáci opravdu využili blok funkce z kategorie start, čímž se program pochopitelnější a budou se v něm dále moci lépe orientovat. Poté co žáci vytvoří funkční pohyb robota, pomocí pohybových bloků a cyklů, tak je čeká, aby se zamysleli, jak by mohli program pozměnit pro více obecné řešení. Jedná se znovu o využití proměnné, ale tentokrát je již program trochu složitější.

Cílem této úlohy je v žácích rozvíjet schopnost tvořivosti, ale také vytrvalosti, pokud se jim na první pokus program nepovede vytvořit. Dále tato úloha zlepší jejich algoritmické myšlení a schopnost orientace v prostoru. Neméně důležitým aspektem této úlohy je první seznámení žáků s blokem IF a ELSE, proto žáci musí být s podmínkami již z dřívějších seznámení, abychom zamezili možným problémům, kdy žáci nebudou vědět, jak s blokem pracovat. Případně můžeme před touto úlohou jakožto učitel žákům tento blok dodatečně vysvětlit.

## Vzorové řešení:



V základní části této úlohy si žáci vyzkouší vytvoření vlastní funkce, se kterou mohou následně lépe pracovat. Důležité je tedy správné užití bloku Function a následné zasazení do programu. Také by si měli všimnout různých opakování a program tedy pomocí cyklu zkrátit. Funkcím by bylo vhodné věnovat více času, protože se mohou hodit i v dalších úlohách a práci poměrně značně zjednoduší.



V rozšíření této úlohy si žáci znovu vyzkouší využití proměnných a vytvořit jednoduché rozhodování robota. Znovu bych rozšíření volil spíše pro rychlejší žáky, kteří již budou mít správně vyřešenou základní část úlohy. V této úloze se již využívají současně podmínky a proměnné, proto může být řešení poměrně problematické.

## 6.2 Spolupráce robotů

Následující úlohy jsou zaměřeny hlavně na vzájemnou spolupráci obou robotů. Je zde již více využíván Dot a jeho nejrůznější vstupy jako jsou například tlačítka nebo čidla náklonu. Úlohy jsou znovu seřazeny podle náročnosti a postupně jsou přidávány nebo opakovány nejrůznější znalosti jako například využití opakování, podmínek a v neposlední řadě i proměnných. Před vstupem do této části úloh je důležité, aby za sebou žáci již měli správně splněnou alespoň většinu z předchozího bloku úloh.

### **6.2.1 Reagujeme na Dota**

#### **Zadání:**

V těchto úkolech se již zaměříme více na spolupráci obou robotů čili budeme Dota využívat více aktivněji než v minulém bloku úloh. Nejdříve si vyzkoušíš, jak může Dash reagovat na Dota.

Tvým hlavním úkolem bude vytvořit program, díky kterému se Dash začne pohybovat vpřed. Pomocí bloků zajisti, aby v případě, že bude zmáčknuto prostřední tlačítko na Dotovi, Dash se okamžitě zastaví.

**Rozšíření:** Místo pohybu vpřed využij pohyb do čtverce.

#### **Cíle a poznámky k úloze:**

Reagujeme na Dota je první úloha z části spolupráce robotů, žáci se naučí používat oba roboty současně a pomocí Dota ovládat Dashe. V této úloze mají žáci za úkol vytvořit program tak, aby mohli pomocí tlačítka na Dotovi Dashe zastavit. Cílem této úlohy je hlavně žákům ukázat, že mohou k dosažení stanoveného cíle využívat i interakce mezi roboty. Žáci se naučí využívat vstupy jako prostředek k řízení a chování robotů. Co se týče rozšíření, tak jde pouze o zopakování pohybových bloků pro budoucí programování.

### Vzorové řešení:



V této úloze se žáci začnou postupně seznamovat s Dotem. Důležité je, aby správně užíli cykly a zároveň si osvojili využití některého vstupu od Dashe.

## 6.2.2 Kapitán Dot

### Zadání:

V této úloze bude Dot v roli šéfa a bude Dashovi udávat směr pohybu.

Hlavním úkolem bude vytvořit program, díky kterému budeš moci tím, že položíš Dota hlavou dolů rozpohybovat Dashe směrem vpřed.

**Rozšíření:** Pokud již Dash zvládá pohyb vpřed, můžeš Dotovi přidat více funkcí. Konkrétně vytvoř takový program, aby se Dash pohyboval podle toho, jak bude Dot zrovna nakloněn.

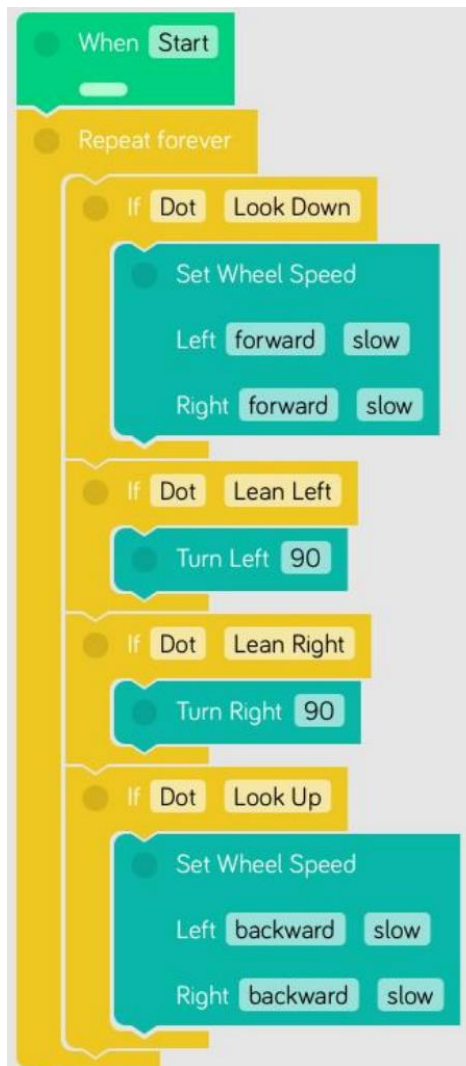
Dash se bude pohybovat tímto způsobem:

- Dot má obličej směrem dolů – Dash se pohybuje dopředu
- Dot je na levém boku – Dash se otočí o 90° doleva
- Dot je na pravém boku – Dash se otočí o 90° doprava
- Dot je hlavou vzhůru – Dash couvá

### Cíle a poznámky k úloze:

Úloha s názvem Kapitán Dot se zaměřuje hlavně na využití funkce IF. Nejdříve žáci použijí pouze jednu podmínku, díky které rozpohybují Dashe. Následně jsou díky rozšíření nuceni řetězit několik podmínek za sebe tak, aby se robot choval podle zadání. Hlavním cílem je tedy u žáků rozvíjet znalost využití podmínek se kterou se již setkali v předchozích úlohách, ale tentokrát bez využití proměnných. Důležité je to, aby žáci nastavili nekonečné opakování, čímž zajistí, že bude program fungovat po celou dobu od spuštění, a ne pouze na využití jedné podmínky. Dále je tato úloha velmi interaktivní, čímž by mohla být pro žáky velmi zábavná a poučná.

### Vzorové řešení:



V tomto řešení je důležité, aby si žáci uvědomili, že musí využít nekonečný cyklus, do kterého následně vnoří podmínky určené zadáním. Není žádoucí, aby v této úloze žáci používali blok When.

### 6.2.3 Zkušený trenér Dot

#### Zadání:

Představ si, že je Dot trenérem a snaží se Dashe připravit na atletický závod. Tentokrát mají trénink běhu a Dot určuje rychlost.

Tentokrát budeme zase pracovat s tlačítky na Dotovi. Tvým úkolem bude zajistit, aby se Dash pohyboval dopředu nebo couval. Ovládat jeho pohyb ale bude Dot pomocí svých tlačítek. Dashův pohyb by se měl měnit tímto způsobem.

- stisk 1. tlačítka – Dash zrychlí
- stisk 2. tlačítka – Dash zpomalí
- stisk 3. tlačítka – Dash se úplně zastaví

Smyslem hry je tedy to, aby Dash jezdil a upravoval svou rychlost na základě pokynů z Dotových tlačítek.

Tuto úlohu řádně otestuj a pokus se zodpovědět tyto otázky:

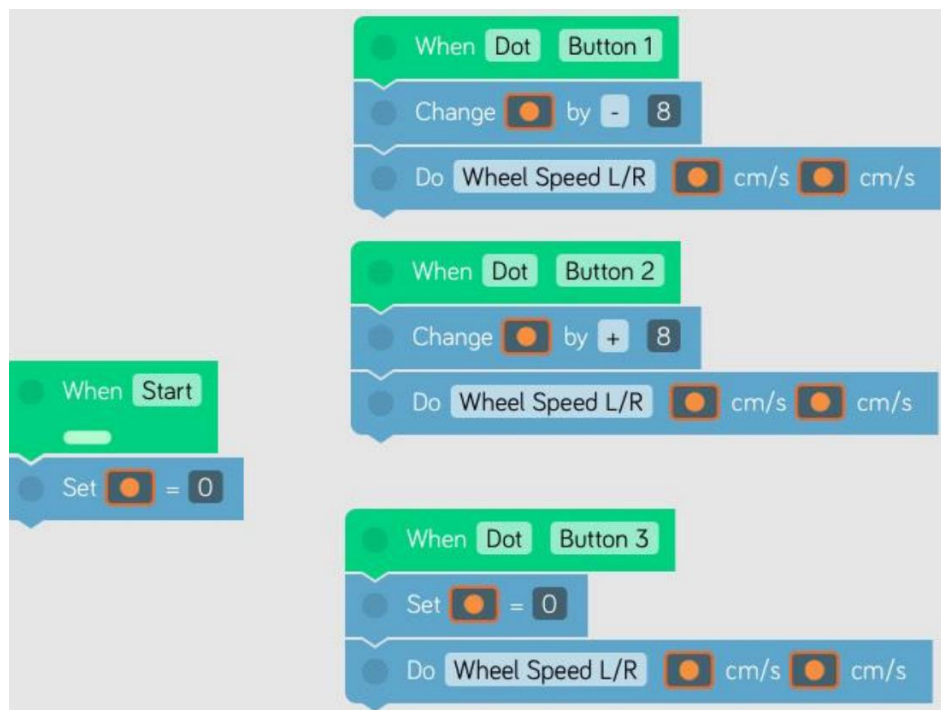
- Mění se rychlost pomocí stejně velkých kroků?
- Pokud Dash stojí, po stisku 1. tlačítka se znovu rozjede pomalou rychlostí?
- Pokud Dash stojí, po stisku 2. tlačítka začne couvat?

Měl by si být schopný si na všechny otázky zodpovědět pozitivně.

#### Cíle a poznámky k úloze:

Poměrně důležitým cílem úlohy Zkušený trenér Dot je u žáků procvičit práci s proměnnými. Jejich úkolem je vytvořit program díky kterému budou po celou dobu programu schopni tlačítka měnit její hodnotu. Konkrétně tak, aby dosáhli požadovaného cíle a měnili Dashovu rychlost. Proto si musí uvědomit co vlastně znamená rychlost v centimetrech za sekundu a že tuto rychlost mohou měnit pomocí čísel. Dále je také důležité úlohu dostatečně odzkoušet a využít správné hodnoty, aby bylo zrychlování co nejefektivnější. Dále se naučí identifikovat a řešit problémy vzniklé během jejich práce. Úloha poskytuje příležitost k praktickému testování a ladění programu v reálném prostředí. Budou rozvíjet schopnost definovat a aktualizovat proměnné podle potřeby a využívat je k uchování nejrůznějších stavů a informací potřebných k vyřešení zadaného problému. Tato úloha také pomocí 47 otázek u žáků rozvíjí schopnost sebereflexe a díky těmto otázkám si mohou sami zjistit, zda úlohu splnili správně či nikoliv. Následně tyto informace mohou využít k dalším úpravám svého programu.

### Vzorové řešení:



Zde budou žáci pracovat čistě s proměnnými. Toto řešení je poměrně časově náročné, protože je důležité, aby ho opravdu řádně otestovali. Musí sami vhodně zvolit hodnotu, o kterou budou robota buďto zrychlovat nebo zpomalovat.

## 6.2.4 Předváděčka

### Zadání:

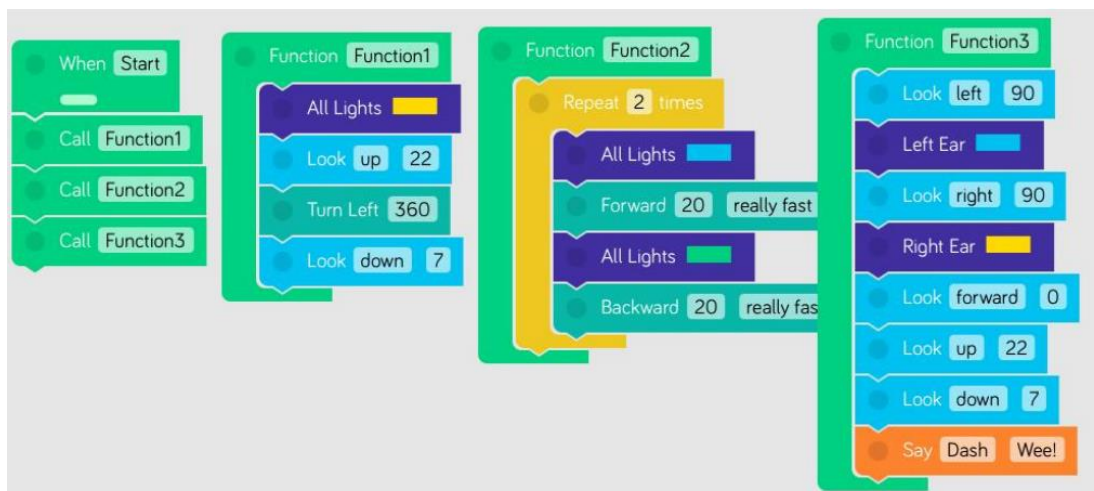
Dash se připravuje na talentové zkoušky a potřebuje oslnit porotu, proto tě poprosil, jestli bys mu nepomohl naučit se nějaké nové triky.

Tvým hlavním úkolem bude vytvořit alespoň 3 Dashovy sestavy. Mělo by se jednat o nějaké zajímavé prvky, které si se během svého dosavadního programování naučil. Může jít například o otočky, různé pohyby hlavou nebo třeba zvukové a světelné efekty. Tvorba sestav je libovolná, jen v každé sestavě použij vždy alespoň 2 z výše zmiňovaných interakcí. Každá reakce by měla být dostatečně originální. Důležité je, abys každou sestavu vytvořil jako funkci a v budoucnu s ní mohl pracovat jako s 1 blokem.

### Cíle a poznámky k úloze:

Předváděčka je ze všech doposud zmiňovaných úloh nejvíce zaměřena na rozvoj kreativity u žáků. Jejich úkolem bude vytvořit alespoň 3 Dashovy sestavy, konkrétně by se mělo jednat o různé otočky, práci s LED světly nebo například mohou využít i zvukové efekty. V této úloze mají žáci pouze tyto podmínky a to, aby byly sestavy dostatečně originální a využívaly alespoň 2 ze zmiňovaných interakcí současně. Z tohoto důvodu není hlavním cílem této úlohy naučit žáky nějaké nové programování, ale spíše jim dát prostor vytvořit něco svého díky doposud získaným vědomostem. Tato úloha je nezbytná pro absolvování následující úlohy, proto je také důležité, aby žáci své výtvořili jako funkce a v příští úloze s nimi mohli pracovat.

### Vzorové řešení:



U tohoto řešení se nemusí kontrolovat nic ohledně využití barev nebo například pohybu. Důležité je, aby žáci splnili podmínku, a to využít alespoň 2 interakce. Dále je důležité animace vytvářet jako funkce pro zjednodušení práce na další úloze.

## 6.2.5 Dot jako dirigent

### Zadání:

Představ si, že je Dot dirigentem a pomocí svých tlačítek bude spouštět některou z předem naučených sestav, které jsi vytvořil v předchozí úloze.

V této úloze budeš znovu pracovat s proměnnými. Tvým úkolem bude tebou vytvořené reakce použít a spouštět pomocí Dota. Reakce se budou spouštěny tímto způsobem:

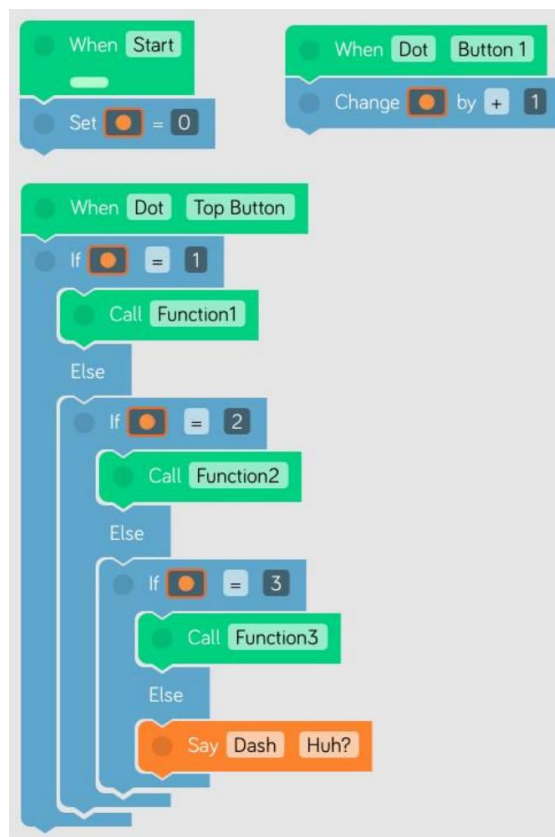
- 1 stisknutí Dotova tlačítka – reakce číslo 1
- 2 stisknutí Dotova tlačítka – reakce číslo 2
- 3 stisknutí Dotova tlačítka – reakce číslo 3

Dále bude potřeba využít ještě jednoho tlačítka, pomocí kterého budeš vybranou sestavu spouštět. Pokud tedy spustíš program, nejdříve naklikáš 1-3x pro výběr sestavy a následně dalším tlačítkem spustíš Dashovu reakci. Pokud bude počet stisknutí jiný než 1, 2 nebo 3, Dash pouze vydá nějaký zvuk a nic nepředvede.

### Cíle a poznámky k úloze:

Co se týče vzdělávacích cílů, tak je úloha Dot jako Dirigent velmi podobná zadání Slalomu. Jedná se také hlavně o práci s vnořenými podmínkami a proměnnými. Tentokrát žáci mají ale jimi vytvořené originální sestavy, což by mohlo vést k většímu zájmu ze strany žáků. Jedná se tedy o využití 2 tlačítek, kdy 1. bude jakési počítadlo, díky kterému žáci namačkají požadovaný počet stisknutí a pomocí 2. tlačítka následně reakci spustí. Pozor bychom si měli dát hlavně na to, aby žáci nevytvořili program, který bude každou reakci spouštět jiným tlačítkem. Tímto by nesplnili zadání, a hlavně by se vyhnuli využití proměnné. Dále je u této úlohy důležité, aby se nespouštěla žádná z reakcí, pokud bude hodnota proměnné jiná, než je 1, 2 nebo 3. Toto by žáci měli vyřešit využitím samostatného zvukového efektu, který bude indikovat, že má proměnná jinou hodnotu.

## Vzorové řešení:



U tohoto řešení je důležité správně zkombinovat již naučené dovednosti, a to konkrétně využití funkcí, rozhodovacích podmínek, nastavování a postupné měnění proměnných. Důležité je vědět, kdy využít blok If a kdy blok When.

### **6.3 Rozšíření**

V následujících úlohách žáci využijí rozšíření. Konkrétně budou pracovat s katapultem, který je součástí robotické sady. Katapult je nutné nejdříve sestavit, proto by bylo vhodné tyto úlohy využívat současně. Všechny 3 úlohy v této části na sebe systematicky navazují a postupně rozvíjí algoritmické myšlení. Žáci si mohou osvojit například využití náhody, opakování, podmínek nebo práci s proměnnými.

### 6.3.1 První výstřely

#### Zadání:

Následující úloha se bude zabývat využitím katapultu a vytvářením programů pro jeho ovládání.

Nejdříve je důležité, abys správně zapojil všechny komponenty, které jsou k tomu potřeba. Použij návod a pokud si nejsi jistý, obrať se na učitele. Dále řádně odzkoušej všechny funkce katapultu, zda funguje správně nabíjení a podobně.

Tvým úkolem bude vytvořit program, díky kterému se Dash bude pohybovat směrem vpřed, dokud před sebou nebude mít nějakou překážku jako je například zeď. Ve chvíli, kdy Dash uvidí překážku, zastaví se a couvne o 60 cm. Následně Dash vystřelí z katapultu směrem k překážce. Nabíjení katapultu v této úloze řešit nemusíš, stačí pokud vždy na začátku Dashovi nabiješ katapult ručně.

#### Cíle a poznámky k úloze:

Úloha s názvem První výstřely je první úloha zaměřující se na využití rozšíření, konkrétně katapultu. Žáci mají za úkol s pomocí učitele správně zapojit všechny důležité části, které jsou nezbytné pro správný chod katapultu. Následně vytvoří program, kde využijí své dosavadní vědomosti ohledně detekování překážek a vyzkouší si samotný katapult.

Tato úloha má několik výukových cílů, konkrétně se zaměřuje na zopakování některých dovedností z minulých úloh a následně na ně navazuje nutností využití nových bloků. U této úlohy a také u všech ostatních úloh je důležité, aby si žáci uvědomili nutnost využití bloku pro čekání, aby vše bezpečně fungovalo. Dále tato úloha pomocí sestavování katapultu rozvíjí motorické schopnosti u žáků.

### Vzorové řešení:



V tomto případě by měla být věnována zvýšená pozornost tomu, zda žáci správně využívají blok Wait, kterým zajistí plynulý chod programu. Dále je také důležité nastavit dostatečnou sílu pro výstřely z katapultu. Záleží pouze na učiteli, zda nechá žáky na problém přijít samotné nebo jim hned na začátku s řešením poradí. Tato úloha se samozřejmě dá řešit i za použití podmínky If, ale toto řešení se mi zdá jednodušší a přehlednější.

### 6.3.2 Střelba na cíl

#### Zadání:

Samotné střílení máš za sebou, tak ho můžeš zakomponovat do složitějších řešení. Do tohoto úkolu se již zapojí i Dot a využiješ nabíjení katapultu.

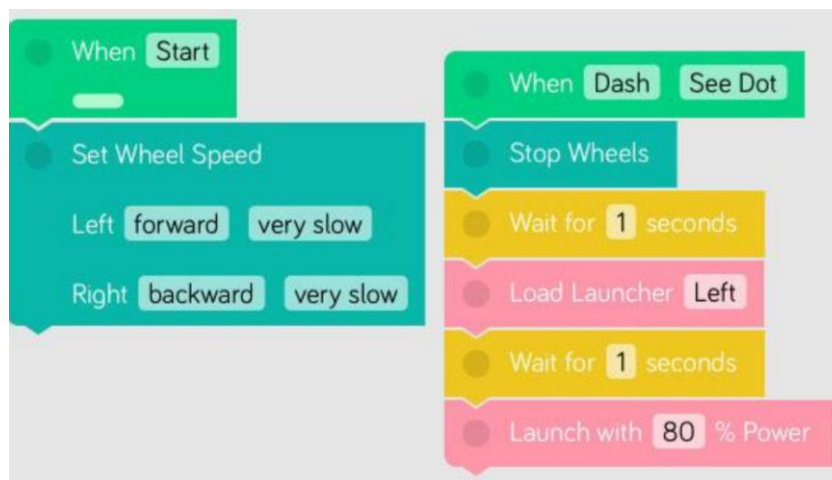
Tvým úkolem bude vytvořit takový program, díky kterému se Dash začne otáčet na jakoukoli stranu. Během chodu programu kdykoliv do prostoru kolem Dashe postav Dota. Ve chvíli, kdy ho Dash uvidí se přestane otáčet. Následně nabije katapult a vystřelí tímto směrem.

#### Cíle a poznámky k úloze:

Co se týče úlohy Střelba na cíl, tak zde žáci vytvoří takový program, aby se Dash otáčel do chvíle, kdy uvidí Dota. Tato úloha se znovu snaží rozvíjet již získané vědomosti ohledně programování pomocí blokového prostředí, ale přidává další rozšíření, a to v podobě nabíjení katapultu. Velmi důležitá část je nastavení samotného otáčení robota. Žáci si musí uvědomit, že mohou robota otáčet i pomocí jiných bloků než těch určených k otáčení. Konkrétně pokud nastaví jedno kolo na směr dopředu a druhé dozadu. Toto je nezbytné k úspěšnému vyřešení úlohy.

Tato úloha cílí na rozvoj představivosti a kreativity. Což je způsobeno hlavně využitím jiného bloku pro otáčení, protože jsou žáci nuceni přijít s jinou alternativou, než na kterou byli doposud zvyklí. V úlohách není schválně uvedena, jakou sílu by měli žáci pro střelbu použít, což je naučí tomu, aby si dokázali s některými problémy poradit samostatně. Konkrétně je problém v tom, že pokud nastaví menší sílu, než je 50 %, tak nedojde k úplnému výstřelu.

### Vzorové řešení:



U této úlohy musí žáci sami přijít na to, že mohou robota otáčet i jiným způsobem než bloky k tomu určenými. Konkrétně tedy tak, že nastaví jedno kolo na otáčení vpřed a druhé kolo do druhého směru, čímž zajistí otáčení do nekonečna.

### 6.3.3 Náhodný katapult

#### Zadání:

V této úloze si zopakujeme cykly a využití proměnné. Z Dashe se v tuto chvíli stane náhodný generátor výstřelů.

Tvým úkolem bude, abys pomocí Blockly vytvořil takovýto program. Dash se otočí o náhodný počet stupňů do náhodné strany, následně pomocí některého ze zvukových efektů dá najevo, že je připraven. V tuto chvíli mu do jednoho z jeho zásobníků připravíš míček. Poté si Dash musí nabít a vystřelit připravený míček tímto směrem. Program se bude opakovat stále dokola, dokud ho sám nevypneš.

#### Cíle a poznámky k úloze:

Náhodný katapult se již podle svého názvu zaměřuje na využití náhodných hodnot. Žáci jsou nuceni robota vždy otočit o náhodný počet stupňů na kteroukoli stranu. Důležité je, aby si uvědomili, že stranu otočení nemusí nikde nastavovat, ale zajistí to správným rozmezím pro náhodnou hodnotu. Konkrétně tak, že v případě záporných hodnot se bude robot otáčet na jinou stranu než v případě hodnot kladných.

Hlavní cíl této úlohy je zopakování několika znalostí. Například o využití cyklu, kdy je v tomto případě nastavit nekonečné opakování tak, aby robot daný pohyb vykonával po celou dobu chodu programu. Dále si žáci zopakují to, že pro správný chod programu musí využít alespoň některá čekání a naposledy si zopakují samotné střílení z katapultu. Na tyto vědomosti následně navazuje využití proměnné a správné dosazení hodnot.

### Vzorové řešení:



U tohoto řešení se znovu jedná o kombinaci několika již známých bloků. Žáci využijí cykly nebo například proměnné a zakomponují to do funkčního programu ovládajícího Dashův katapult. Jedná se o poslední úlohu, proto kombinuje téměř všechny důležité vědomosti z oboru programování.