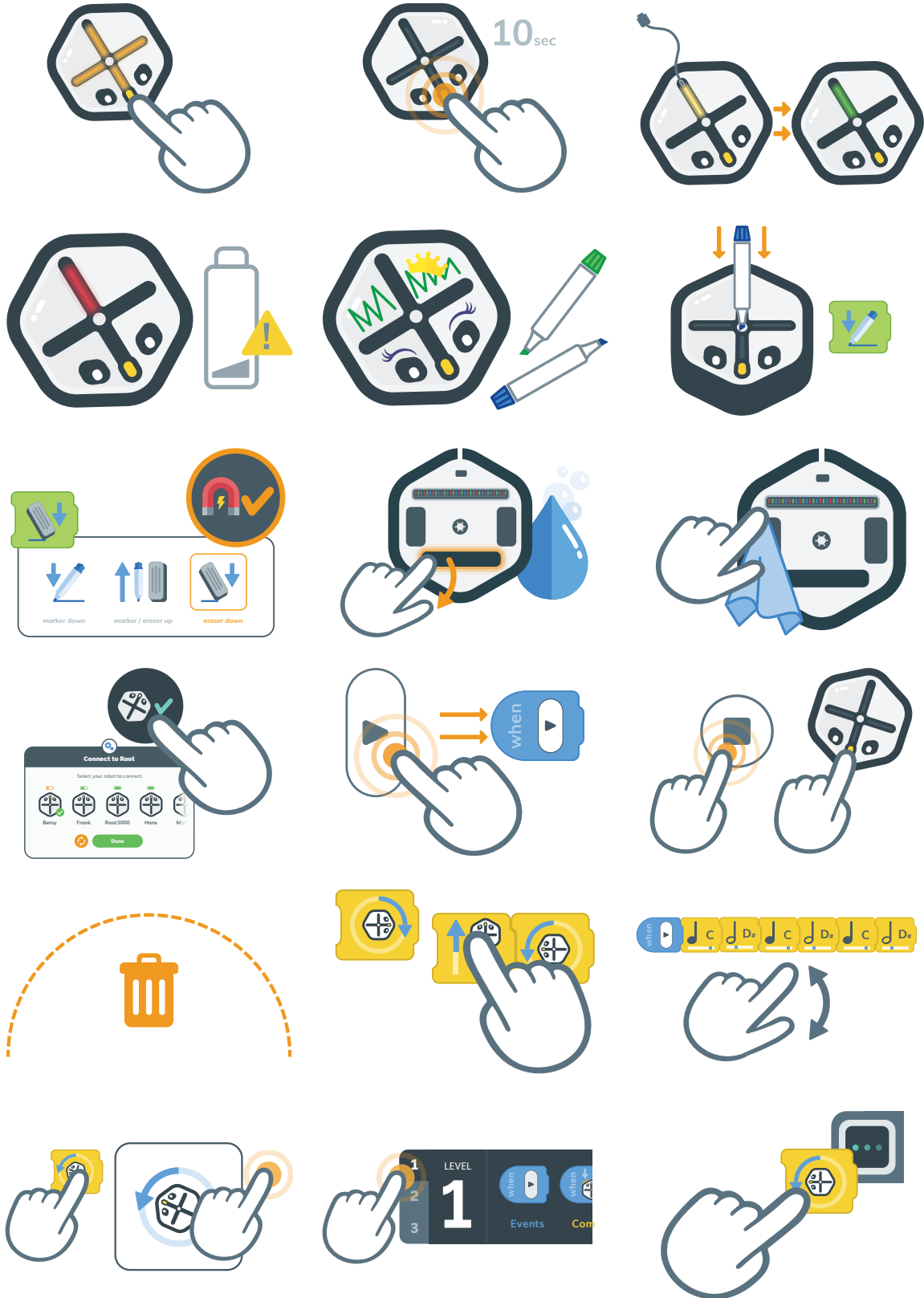
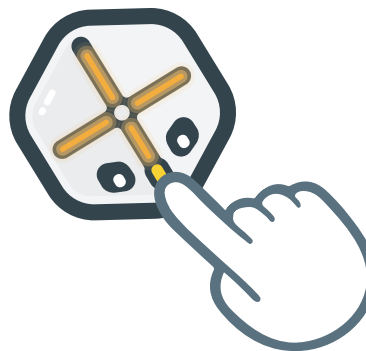


# iRobot Root – příručka kódování



## Zapnutí / vypnutí Roota

Stiskněte a podržte žluté tlačítko (nos) Roota.



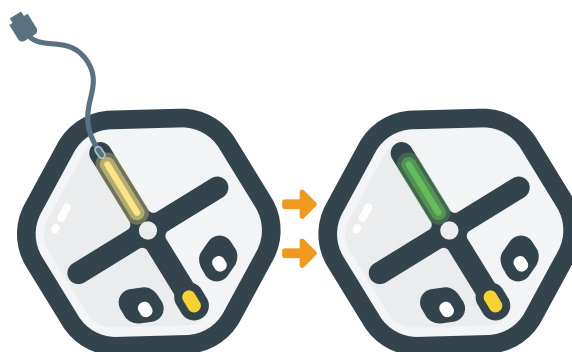
## Resetování Roota

Pokud Root nereaguje nebo se zasekl, podržte žluté tlačítko 10 vteřin stisknuté.



## Nabíjení

Zapojte nabíjecí kabel pro start nabíjení. V průběhu nabíjení bude svítit Root žlutě. Jakmile bude nabitý, bude svítit zeleně.



## Upozornění na vybitou baterii

Root bliká červeně, pokud dochází baterie. Zapojte Roota pro nabití.



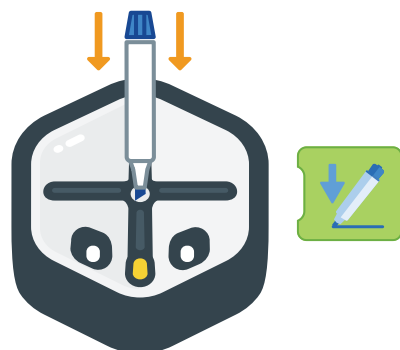
## Zdobení Roota

Vrchní bílá strana Roota je whiteboard, pro zdobení se hodí nálepky nebo smývatelný fix.



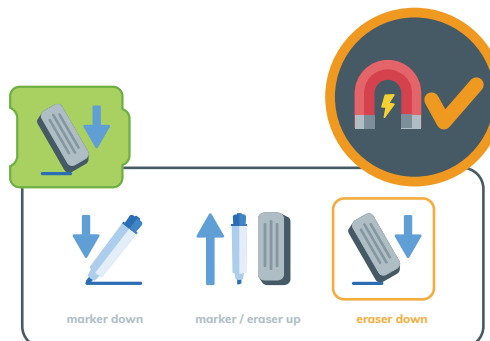
## Malování s Rootem

Pro malování jednoduše zasuňte fix do prostředního držáku. Použijte „Marker block“ pro zvednutí fixu nebo psaní.



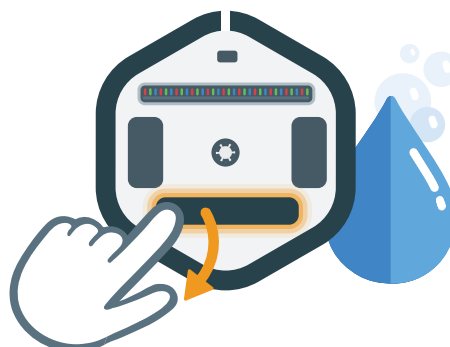
## Mazání s Rootem

Nastavte „Marker block“ na „eraser down“ pro spuštění mazání. Funguje pouze na magnetických tabulích.



## Vyčištění mazací plochy

Vyloupněte mazací plochu a umyjte ji mýdlem a vodou.



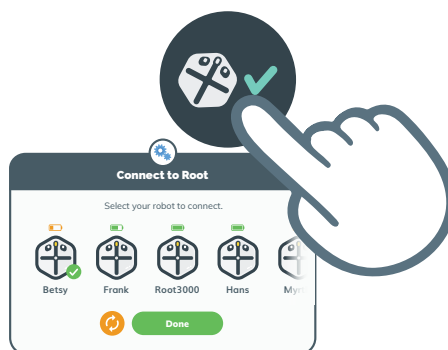
## Čištění kol

Pokud jsou kola Roota špinavé, vyčistěte je hadříkem.



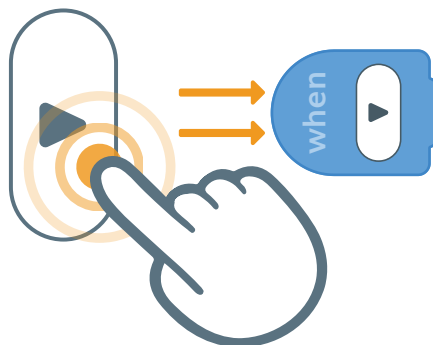
## Připojení Roota

Pro připojení robota klikněte na ikonu Rootu a vyberte svého robota.



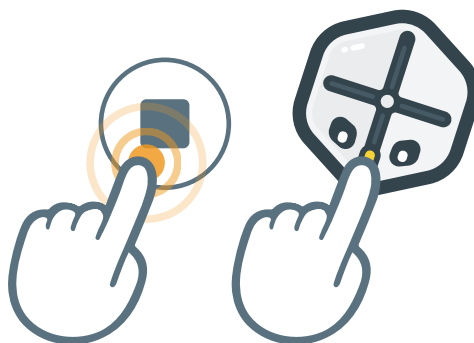
## Spuštění projektu

Stiskněte tlačítko „Play“ v levé části obrazovky pro spuštění projektu. Každý projekt začíná blokem Spustit.



## Zastavení projektu

Pro zastavení běžícího projektu zmáčkněte žluté tlačítko Roota nebo tlačítko Stop v levé stráně obrazovky.



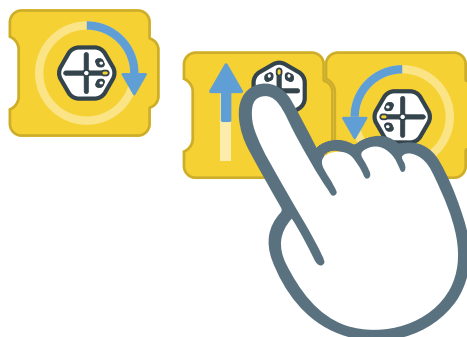
## Mazání bloků

Když přetahujete bloky, ve spodní části obrazovky se objeví koš. Bloky můžete smazat přetáhnutím do koše.



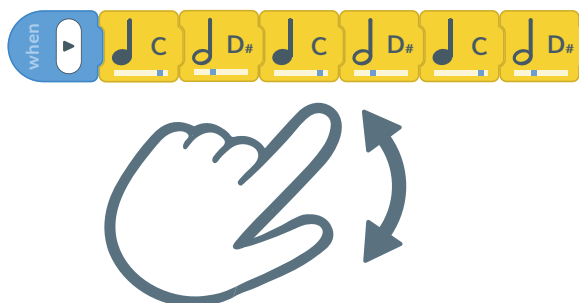
## Přetáhnutí více bloků najednou

Stisknutím a držením bloku můžete přesunout všechny bloky napravo najednou.



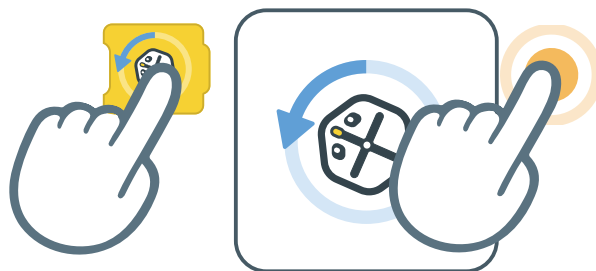
## Zvětšení a zmenšení náhledu na obrazovce

Dva prsty k sobě nebo od sebe pro oddálení nebo přiblížení kódu.



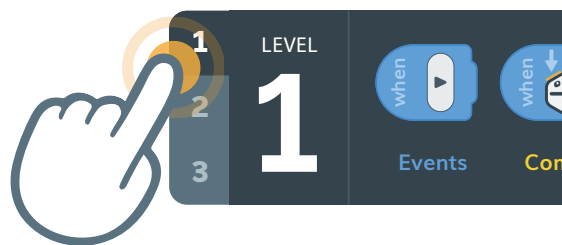
## Otevírání a zavírání editoru bloků

Klikněte na blok pro zobrazení editoru. Klikněte kdekoli mimo okno editoru pro jeho zavření.



## Změna úrovně programování

Kliknutím na číslice měníte úroveň kódování. Pokud nemůže být kód převeden do jiné úrovně, objeví se upozornění.

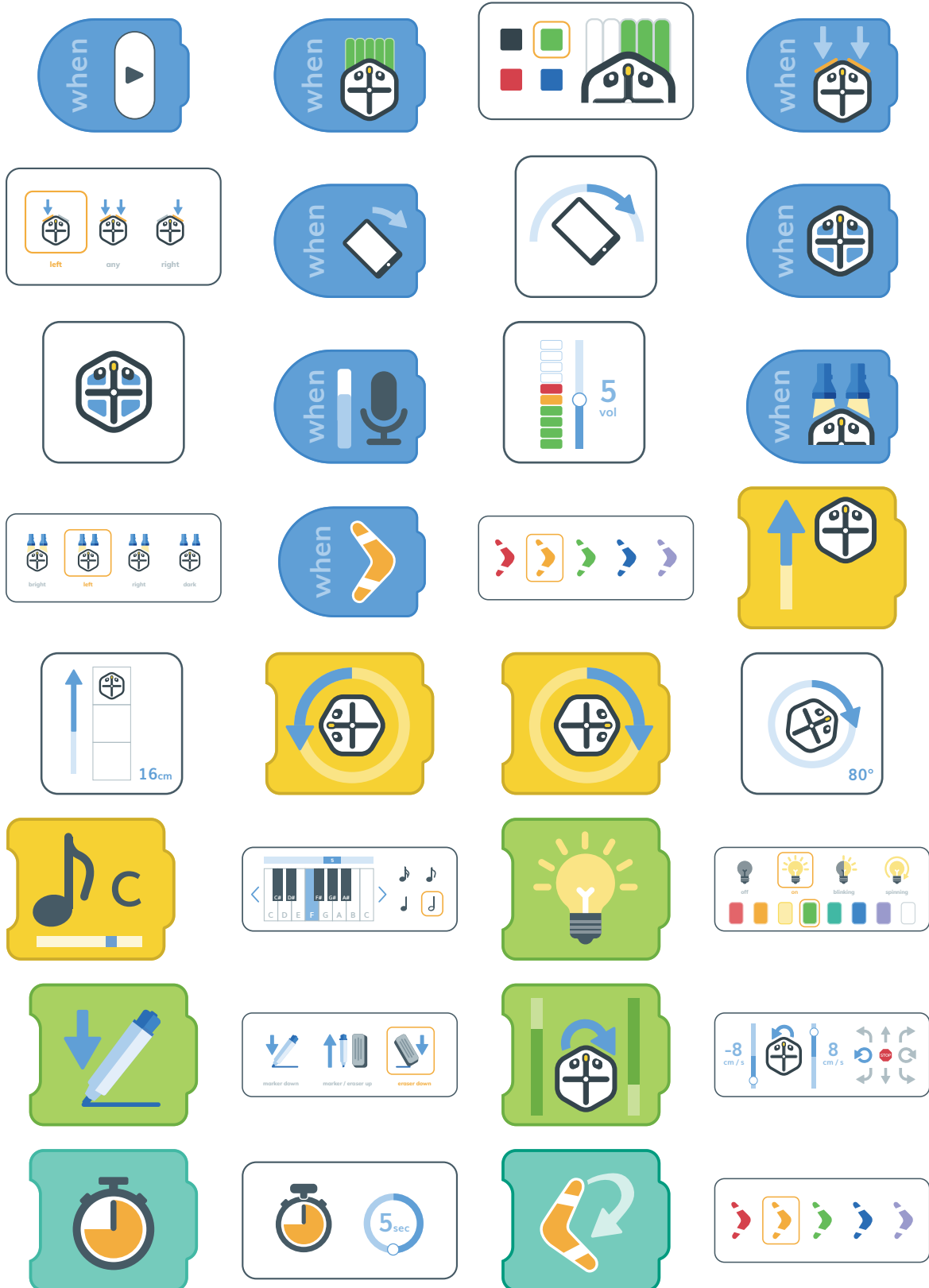


## Adresář a popis bloků

Přetáhněte vybraný blok do ikony Čtverce a dozvíte se podrobnosti o tomto bloku.



# Seznam bloků 1. úrovně



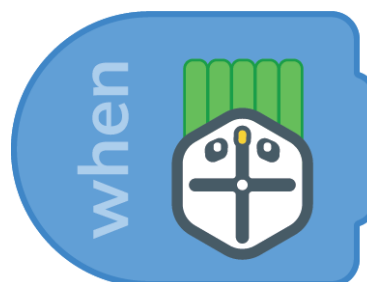
## Spustit

Kód umístěný za tento blok se spustí, jakmile kliknete na tlačítko „Play“.



## Detekce barvy

Tento blok použijte pro detekci barev a reakci Roota na barvu, kterou zvolíte.



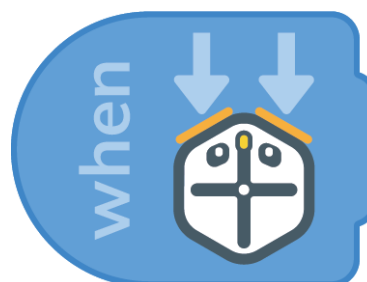
## Nastavení detekce barvy

V editoru nastavte barvy a zóny, které má Root číst.



## Detekce nárazu

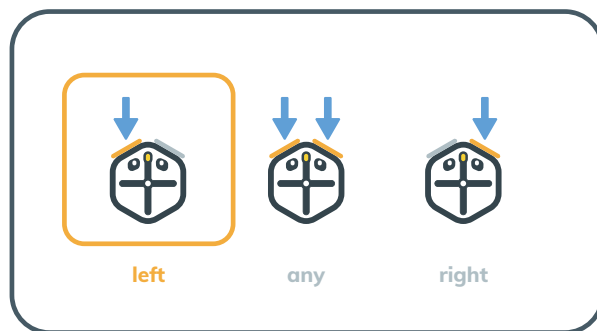
Přední nárazníky Roota mohou být nastaveny jako výchozí akce, která spouští další program.





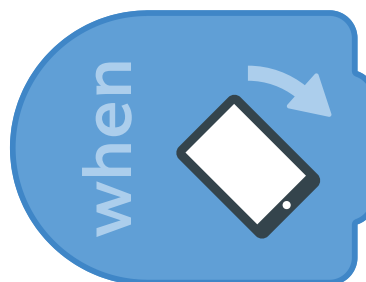
## Nastavení detekce nárazu

Nastavte, na který z nárazníků má  
Root reagovat.



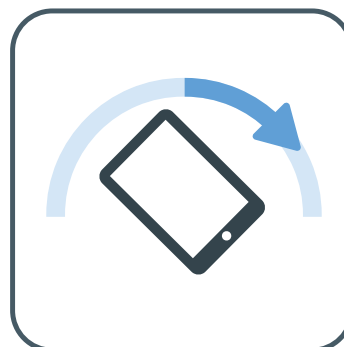
## Detekce náklonu

Blok umožňuje řídit Roota  
pomocí náklonu zařízení  
s iOS.



## Nastavení detekce náklonu

Přetáhněte šipku pro určení  
směru náklonu, podle kterého  
bude Root reagovat.



## Detekce dotyku

Nastavte Roota, aby reagoval  
na stisknutí jedné ze čtyř  
horních dotykových ploch.



## Nastavení detekce dotyku

V editoru můžete určit, které dotykové plochy reagují na dotyk, a které ne.



## Detekce zvuku

Root umí prostřednictvím zařízení s iOS rozpoznávat a reagovat na změny v hlasitosti.



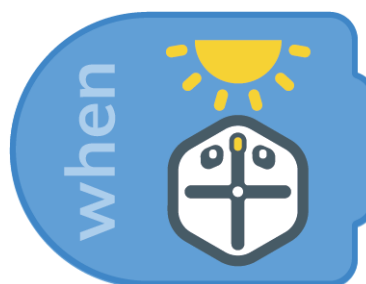
## Nastavení detekce zvuku

Nastavte úroveň hlasitosti, na kterou má Root reagovat.



## Detekce osvětlení

Prostřednictvím zařízení s iOS má Root možnost reagovat na změny světla.



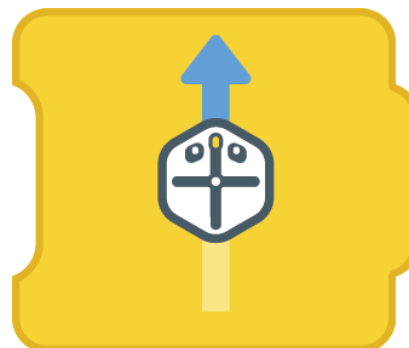
## Nastavení detekce osvětlení

Root má dva světelné senzory, které mohou reagovat na změnu intenzity světla.



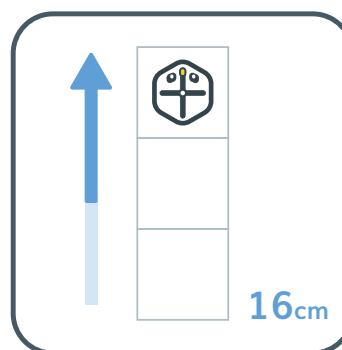
## Posun

Umožňuje pohyb Roota dopředu a dozadu. Můžete nastavit, jakou vzdálenost urazí.



## Nastavení posunu

Přesuňte Roota v editoru pro určení, jak daleko se má pohybovat. Maximální hodnota v první úrovni je 16 cm.



## Otočení

Nastavte otáčení Roota proti směru hodinových ručiček.



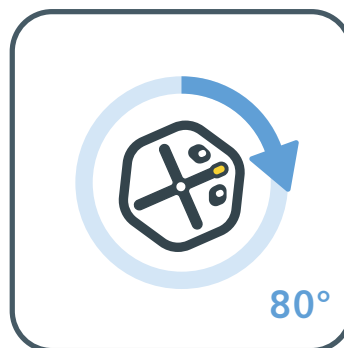
## Otočení

Nastavte otáčení Roota ve směru hodinových ručiček.



## Nastavení otočení

Nastavte úhel, o který se má Root otočit.



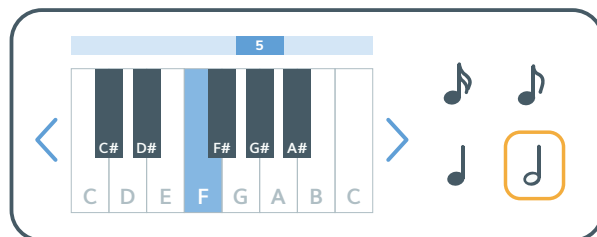
## Melodie

Díky tomuto bloku můžete nechat Roota hrát různé noty. Měňte noty a dobu jejich trvání pro složení písničky.



## Nastavení melodie

Editujte tóny a délky not. Pomocí šipek vlevo a vpravo měňte oktávy.



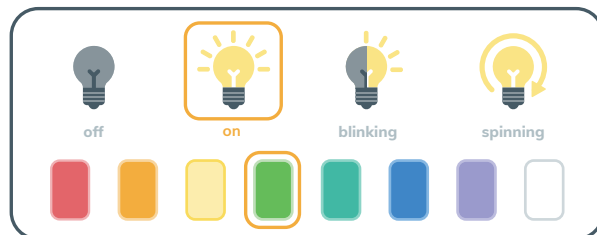
## Svícení

Umožňuje rozsvítit nebo rozblíkat světlo na vrchní straně Roota, měnit jeho barvu nebo změnit styl svícení.



## Nastavení svícení

Použijte editor ke změně barvy a vzorce pro svícení.



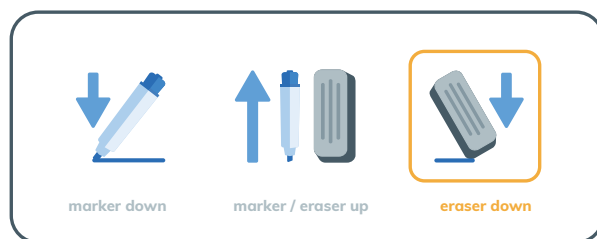
## Popisovač

Tento blok se používá pro zasunutí a vysunutí fixu a mazací plochy.



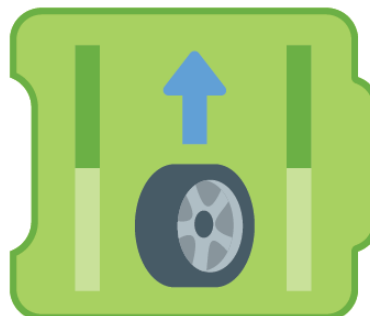
## Nastavení popisovače

Použijte editor pro určení, má-li se Root připravit ke psaní nebo k mazání.



## Rychlost kol

Nastavte rychlost kol pro změnu rychlosti a pro změnu poloměru otáčení Roota.



## Nastavení rychlosti kol

Změňte přímo rychlost a směr každého z kol nebo v pravém editoru nastavte např. otáčení na místě nebo otáčení obloukem.



## Čekání

Blok umožňuje vytvořit časové zpoždění před pokračováním na další blok.



## Nastavení čekání

Nastavte, jak dlouho má Root čekat.



## Opakování

Všechny bloky uvnitř tohoto bloku budou opakovány.

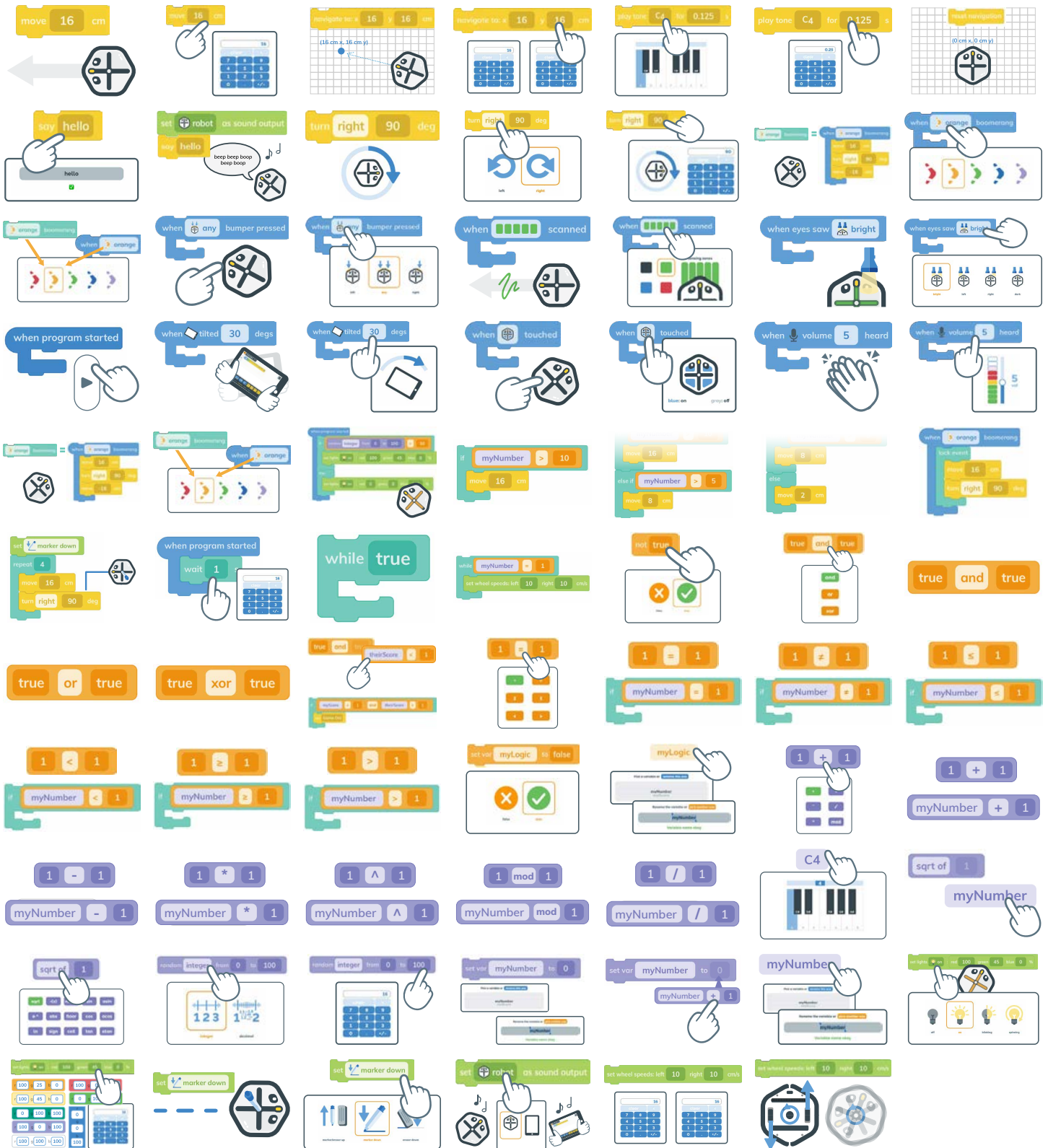


## Nastavení opakování

Nastavte kolikrát se má opakovat kód ve smyčce.



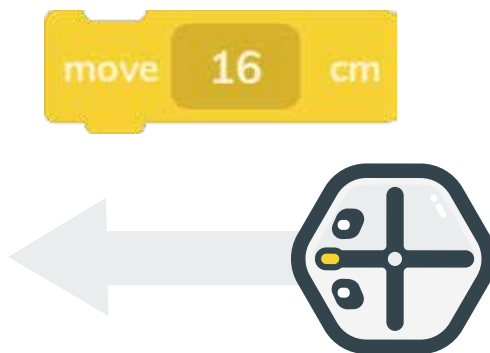
## Seznam bloků 2. úrovně





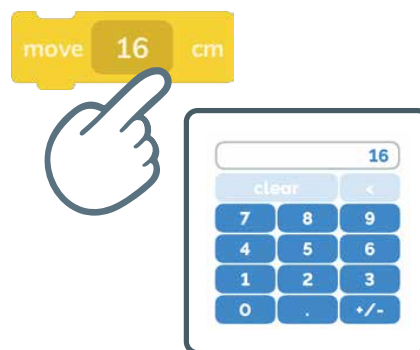
## Posun

Umožňuje pohyb Roota dopředu a dozadu v centimetrech.



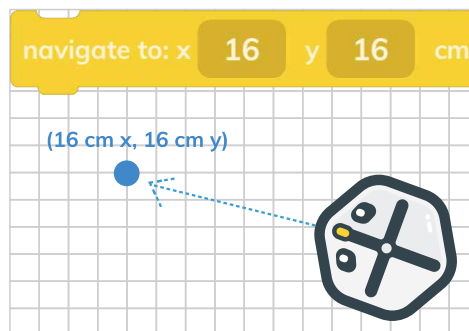
## Nastavení posunu

V editoru zadejte kolik centimetrů má Root urazit dopředu nebo dozadu.



## Navigace

Root se může pohybovat po neviditelné čtvercové síti podle souřadnic os x a y – startovní pozice Roota je 0 cm, 0 cm.



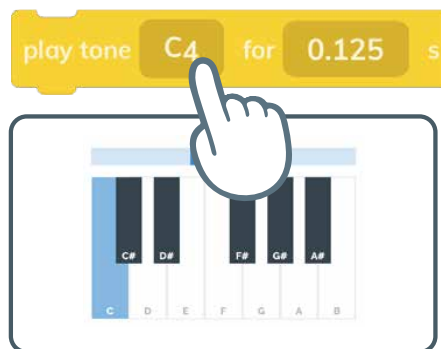
## Nastavení navigace

Navigujte Roota na specifickou pozici zadáním souřadnic do editoru.



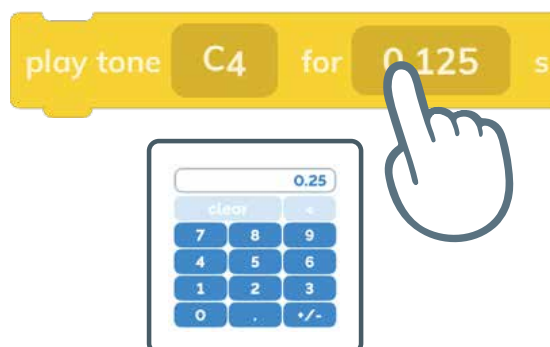
## Melodie

Díky tomuto bloku můžete nechat Roota hrát různé noty. Otevřete první editor pro výběr noty.



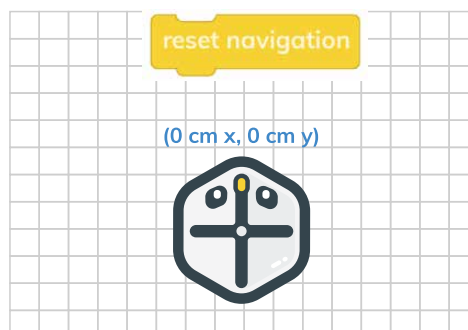
## Nastavení melodie

Otevřete druhý editor a určete, kolik vteřin má Root hrát vybranou notu.



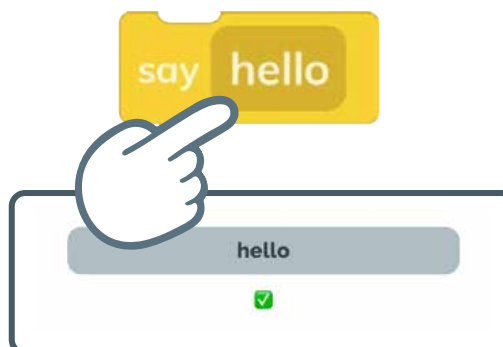
## Nulování navigace

Tento blok nastaví aktuální pozici Roota na neviditelném čtvercovém poli jako pozici startovní (0 cm, 0 cm).



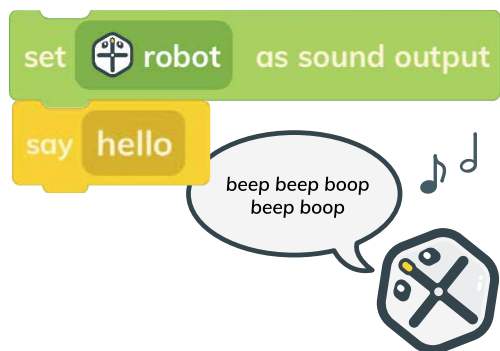
## "Mluv"

Nastavte blok, aby Root nebo vaše zařízení "řeklo" něco nahlas.



## Jazyk Roota

Pokud budete kódovat Roota, aby mluvil, Root si slova přeloží do svého jazyka.



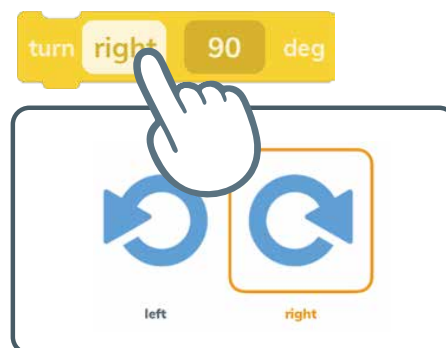
## Otočení

Nastavte otáčení Roota vpravo nebo vlevo.



## Nastavení otočení

Použijte první pole pro určení směru otočení.



## Nastavení otočení

Použijte druhé pole pro určení úhlu otočení.



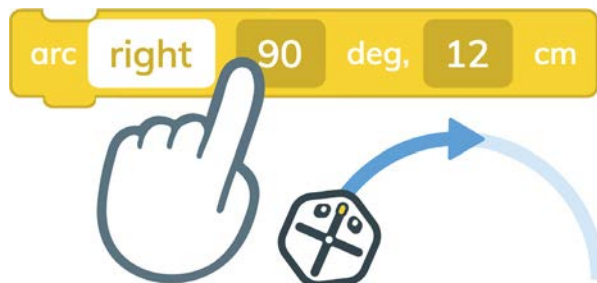
## Oblouk

Tento blok řídí Roota podle tvaru kruhu.



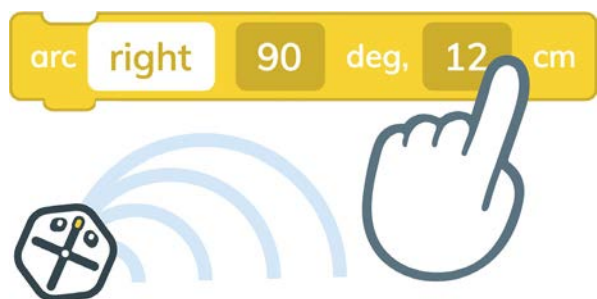
## Nastavení oblouku

Úhel v prvním poli říká Rootu, jak daleko má jet po kružnici.



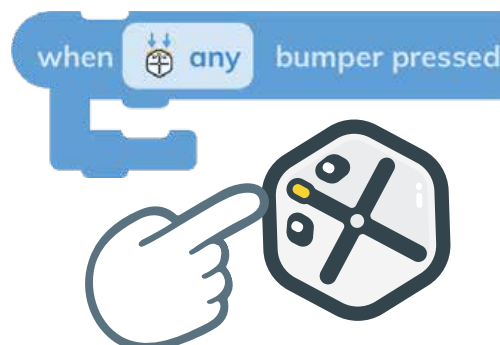
## Nastavení oblouku

Rádus bloku oblouk říká Rootu, jaký je poloměr kružnice, po které má jet.



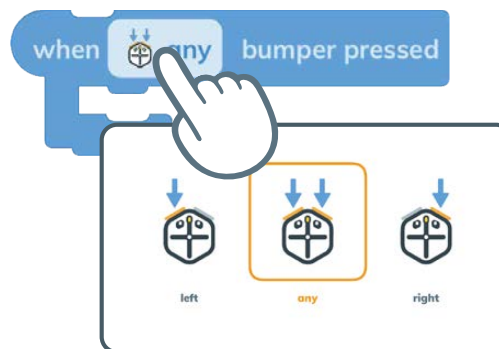
## Detekce nárazu

Přední nárazníky Roota mohou být nastaveny jako výchozí akce, která spouští další program.



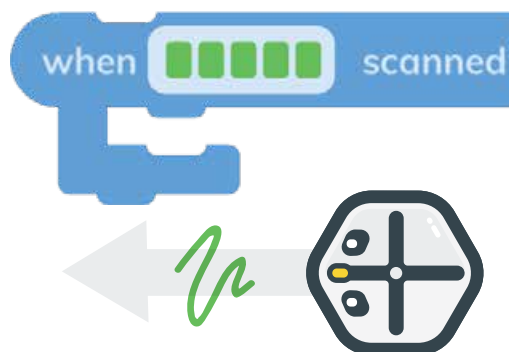
## Nastavení detekce nárazu

Nastavte, na který z nárazníků má Root reagovat.



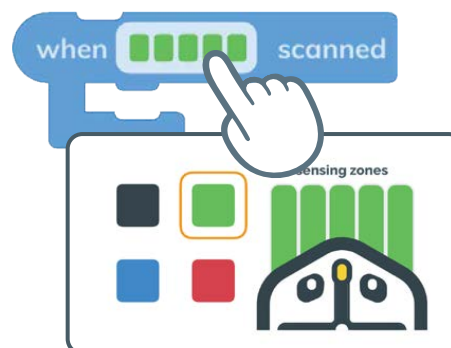
## Detekce barvy

Tento blok použijte pro detekci barev a reakci Roota na barvu, kterou zvolíte.



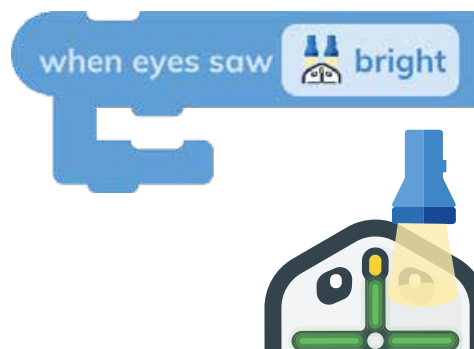
## Nastavení detekce barvy

V editoru nastavte barvy a zóny, které má Root číst.



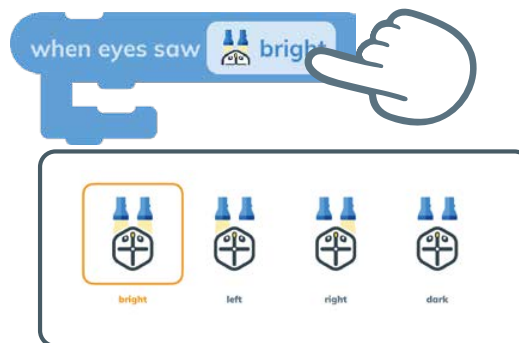
## Detekce osvětlení

Prostřednictvím zařízení s iOS má Root možnost reagovat na změny světla.



## Nastavení detekce osvětlení

Root má dva světelné senzory, které mohou reagovat na změnu intenzity světla.



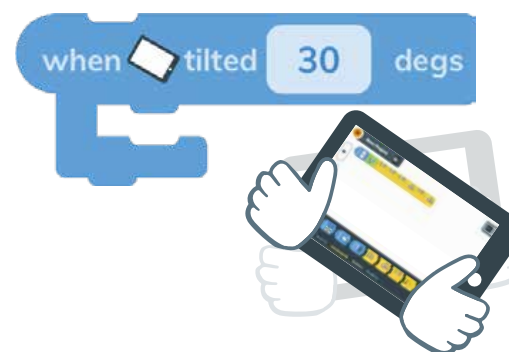
## Spustit

Kód umístěný za tento blok se spustí, jakmile kliknete na tlačítko start.



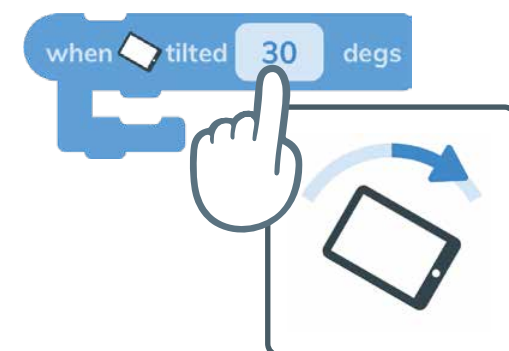
## Detekce náklonu

Blok umožňuje řídit Roota pomocí náklonu zařízení.



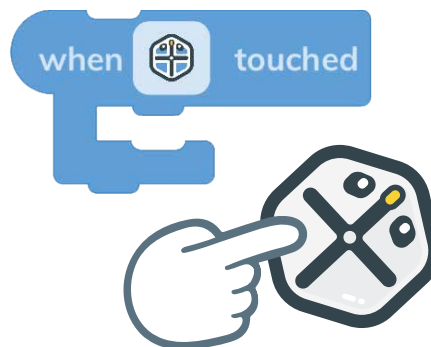
## Nastavení detekce náklonu

Přetáhněte šipku pro určení úhlu náklonu, podle kterého bude Root reagovat.



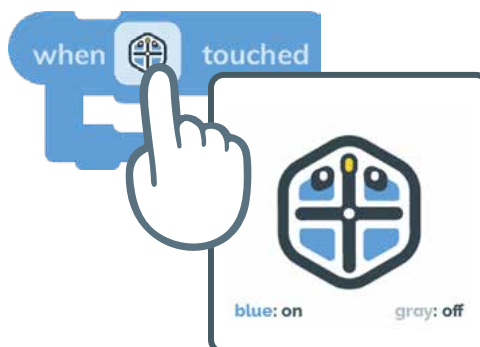
## Detekce dotyku

Nastavte Roota, aby reagoval na stisknutí jedné ze čtyř horních dotykových ploch.



## Nastavení detekce dotyku

V editoru můžete určit, které dotykové plochy reagují na dotyk, a které ne.



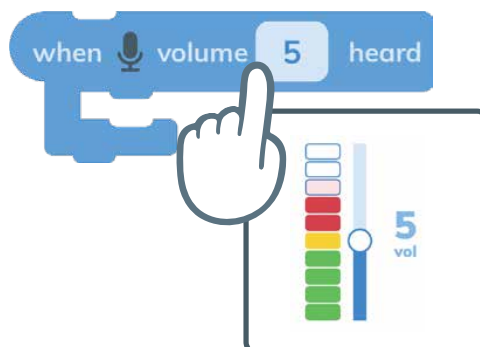
## Detekce zvuku

Root umí prostřednictvím zařízení s iOS rozpoznávat a reagovat na změny v hlasitosti.



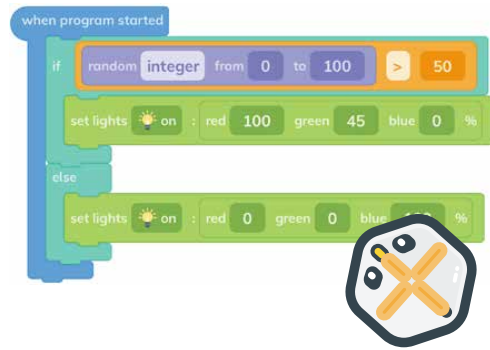
## Nastavení detekce zvuku

Nastavte úroveň hlasitosti, na kterou má Root reagovat.



## Podmínky

Bloky podmínek říkají Rootu, co má udělat, POKUD je nějaká akce PRAVDA nebo NEPRAVDA.



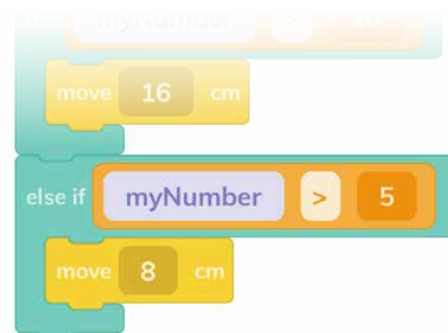
## Pokud (If)

Root bude následovat kód uvnitř tohoto bloku POKUD je výrok uvnitř tohoto bloku PRAVDA. Pokud je NEPRAVDA, program bude ignorovat tento blok a bude pokračovat dál.



## Pokud jinak (Else If)

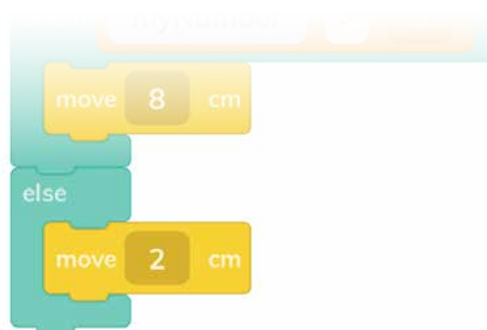
Pokud je výrok uvnitř tohoto bloku Pokud NEPRAVDA program pokračuje na blok Pokud jinak. Pokud je výrok PRAVDA, bude tento kód proveden. Pokud výrok pravdivý není program pokračuje na další blok.





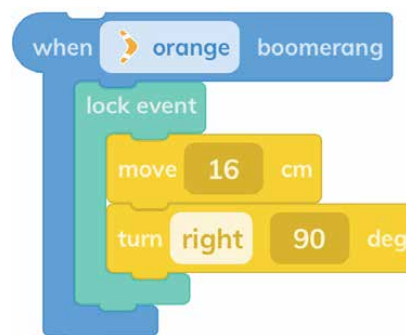
## Jinak (Else)

Pokud je výrok uvnitř bloků  
Pokud a všech bloků  
Pokud jinak neplatný,  
program bloku Jinak je  
spuštěn.



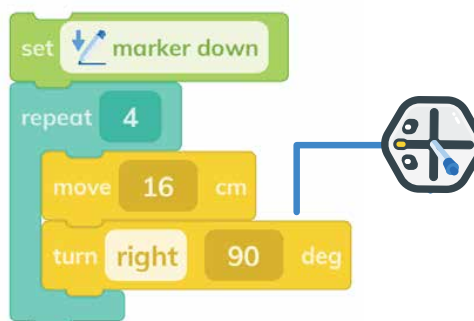
## Uzamčená událost (Lock Event)

Když program Rootu dosáhne  
bloku Uzamčená událost,  
projde celý kód, který tento  
blok obsahuje bez přerušení  
dalšími událostmi.



## Opakování

Všechny bloky uvnitř tohoto  
bloku budou opakovány podle  
stanoveného počtu opakování.



## Čekání

Blok umožňuje vytvořit časové zpoždění před pokračováním na další blok.



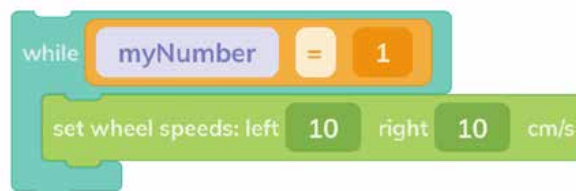
## Dokud

Blok Dokud může nabývat hodnot PRAVDA nebo NEPRAVDA



## Podmínka dokud

Pokud je výrok uvnitř tohoto bloku PRAVDA je tento kód opakován, dokud se výrok nezmění na NEPRAVDU.



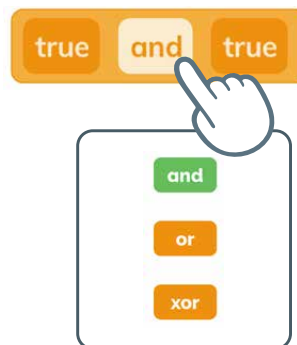
## Negace

Otáčí výrok na jeho opak. PRAVDA se mění na NEPRAVDU a NEPRAVDA na PRAVDU.



## Logické operace

Tento blok může být jak PRAVDA tak i NEPRAVDA v závislosti na stavu jeho obsahu.



### Logický součin (AND)

Blok vrátí PRAVDA (true), pokud jsou pravdivé všechny výrazy v bloku.



### Logický součet (OR)

Blok vrátí PRAVDA (true), pokud je pravdivý alespoň jeden výraz v bloku.



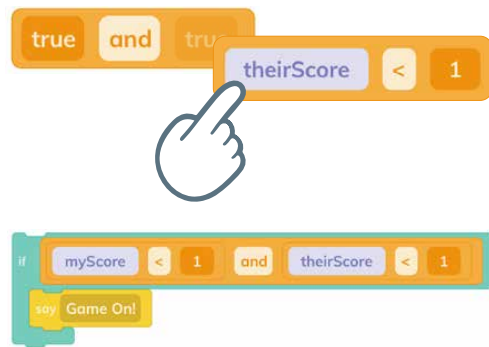
### Exkluzivní logický součet (XOR)

Blok vrátí PRAVDA (true), pokud je pravdivý pouze jeden (a ne oba) výrazy v bloku.



## Používání podmínek

Použijte logické operace s podmínkami, a řekněte tak Rootovi, co má dělat, pokud jsou výrazy uvnitř pravdivé nebo nepravdivé.



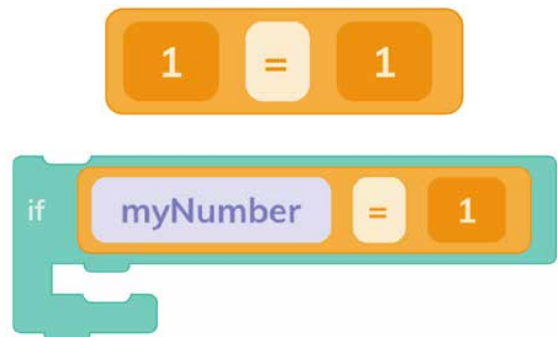
## Porovnání

Srovnávací blok vrátí buď PRAVDA (true), nebo NEPRAVDA (false) v závislosti na vztahu mezi jednotlivými poli.



## Rovná se

Vrací PRAVDA pokud je první hodnota stejná jako hodnota druhá.



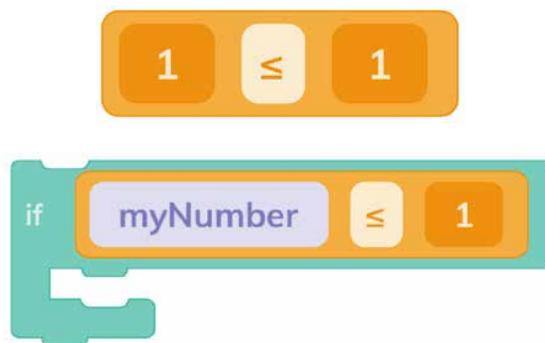
## Nerovná se

Vrací PRAVDA pokud není první hodnota stejná jako hodnota druhá.



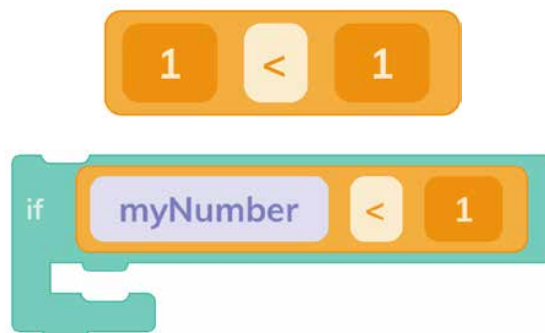
## Je menší nebo rovno

Vrací PRAVDA pokud je první hodnota nižší než hodnota druhá.



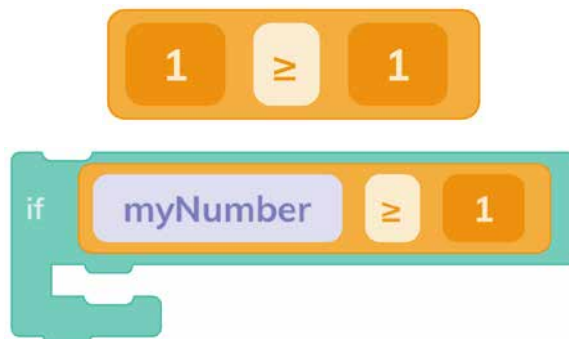
## Je menší než

Vrací PRAVDA pokud je první hodnota nižší než hodnota druhá.



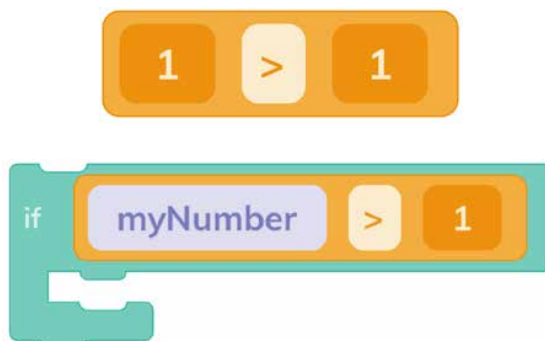
## Je větší nebo rovno

Vrací PRAVDA pokud je první hodnota vyšší nebo stejná oproti hodnotě druhé.



## Je větší než

Vrací PRAVDA pokud je první hodnota vyšší než hodnota druhá.



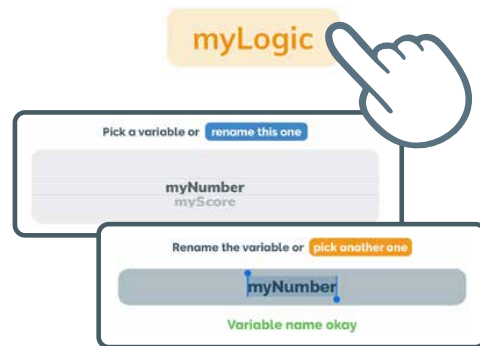
## Nastavení Boolean proměnné

Logickou proměnnou můžete nastavit na PRAVDU (true) nebo NEPRAVDU (false) v závislosti na události, například když klepnete na nárazník.



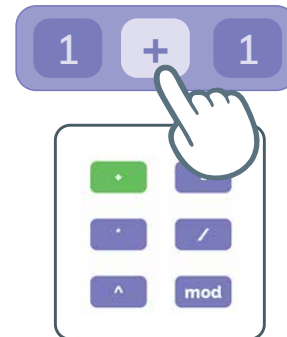
## Boolean proměnná

Logické proměnné mohou přepínat mezi pravdivými nebo nepravdivými hodnotami, aby pomohly Rootovi sledovat, co se děje v projektu.



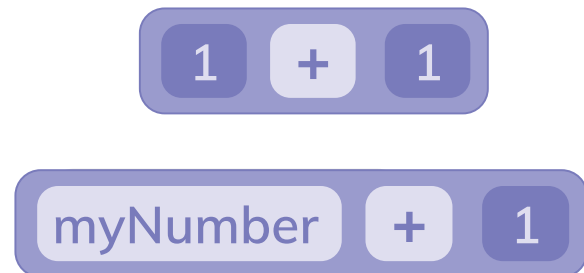
## Výpočet

Zobrazí výsledek vybrané matematické operace.



## Součet

Sčítání dvou hodnot.



## Rozdíl

Odčítání dvou hodnot.



## Součin

Násobení dvou hodnot.



## Mocnina

První číslo je umocněno pomocí hodnoty ve druhém poli.



## Zbytek

Výsledek je roven zbytku po podílu první a druhé hodnoty.



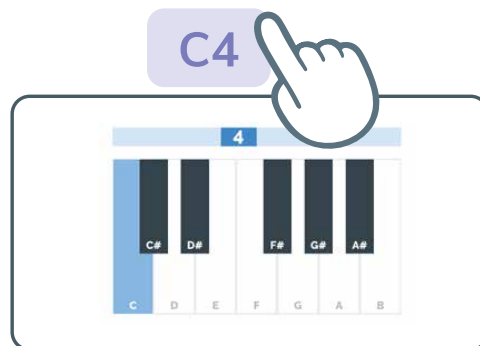
## Podíl

Blok dělení umožňuje dělit dvě zadané hodnoty.



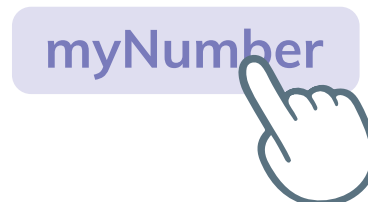
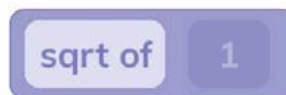
## Frekvence tónu

Umožňuje nastavit zvolenou frekvenci not v Hz.



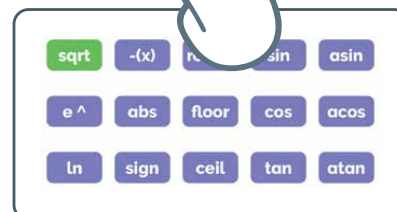
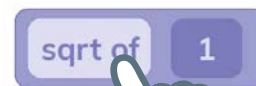
## Matematická operace

Editor umožňuje výběr požadované matematické operace.



## Výběr operace

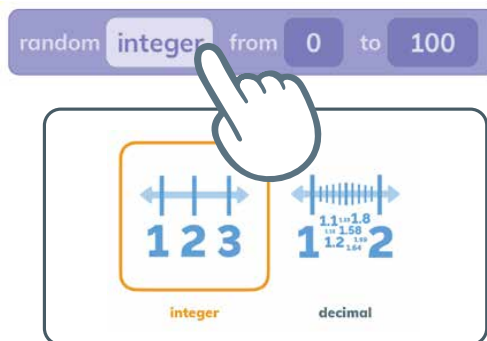
Editor umožňuje výběr požadované matematické operace.





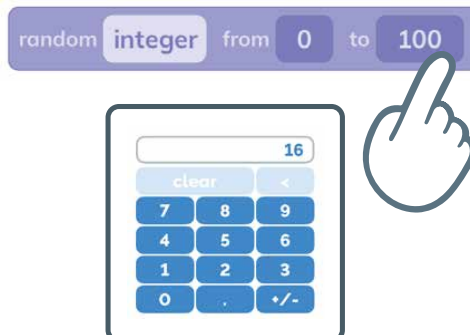
## Náhodné číslo

Tento blok generuje jakékoli celé nebo desetinné číslo uvnitř rozsahu dvou hodnot.



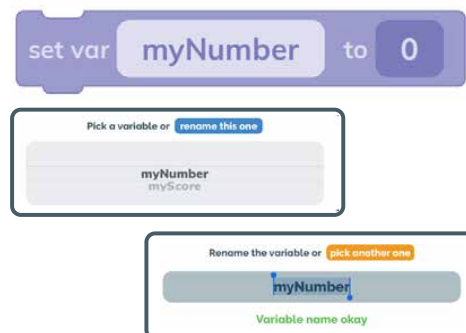
## Rozsah náhodného čísla

Rozsah můžete změnit pomocí dvou číselných polí.



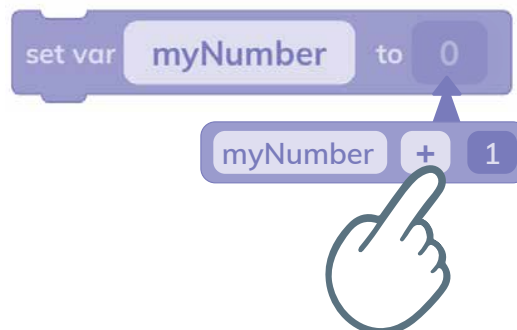
## Nastavení proměnné

Zde nastavíte počáteční hodnotu proměnné.



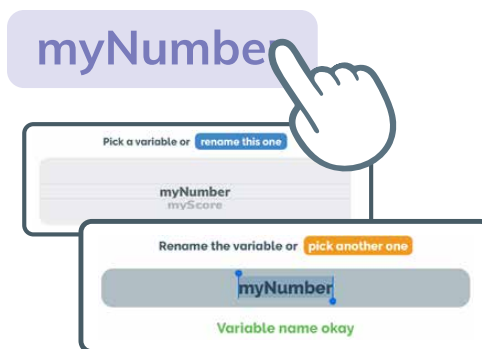
## Změna hodnoty

Blok nastavení proměnné může obsahovat číslo nebo výsledek početní operace. Lze tak například připočítat "1" k aktuální hodnotě proměnné.



## Proměnná

Čísla proměnných udržují hodnoty, které pomáhají Rootu sledovat, co se děje v projektu.



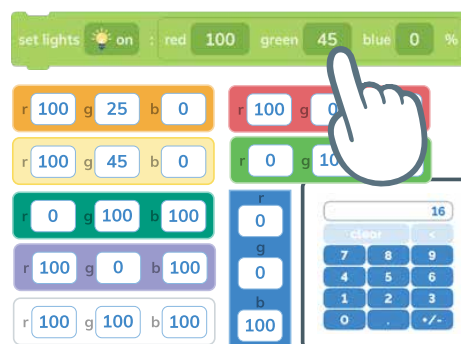
## Svícení

Umožňuje rozsvítit nebo rozblíkat světlo na vrchní straně Roota, měnit jeho barvu nebo změnit styl svícení.



## Nastavení svícení

Použijte editor ke změně barvy a stylu svícení. Všechny barvy světla jsou směsí červené, zelené a modré barvy. Vytvořte různé barvy nastavením různých hodnot R, G, B na různá čísla.



## Popisovač

Tento blok se používá pro zasunutí a vysunutí fixu a mazací plochy.



## Nastavení popisovače

Použijte editor pro určení, má-li se Root připravit ke psaní nebo k mazání.



## Nastavení zvukového výstupu

Nastavte přehrávání zvuků na vašem zařízení s iOS nebo prostřednictvím Roota.



## Rychlost kol

Nastavte rychlost kol pro změnu rychlosti a pro změnu poloměru otáčení Roota.

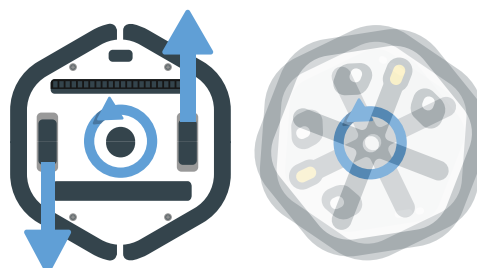
set wheel speeds: left 10 right 10 cm/s



## Nastavení rychlosti kol

Nastavte rychlost jednoho z kol na negativní hodnotu. Jak se bude Root pohybovat?

set wheel speeds: left 10 right 10 cm/s



## Seznam bloků 3. úrovně

**robot.move( 16 )**

**robot.move( 16 )**

**robot.navigateTo( x:16 , y:16 )**

**robot.navigateTo( x:16 , y:16 )**

**play(Tone(freq:16 , duration:16))**

**play(Tone(freq:16 , duration:16))**

**robot.resetNavigation()**

**robot.say( "hello" )**

**robot.say( "hello" )**

**robot.turn( right , 90 )**

**robot.turn( right , 90 )**

**robot.turn( right , 90 )**

**boomerang(orange)**

**robot.whenBoomerang( .orange )**

**robot.whenBoomerang( .orange )**

**robot.whenBumperPressed ([true, true])**

**robot.whenBumperPressed ([true, true])**

**robot.whenColorScanned ([color, color, color, color, color])**

**robot.whenColorScanned ([green, green, green, green, green])**

**robot.whenEyesSee(.bright)**

**robot.whenEyesSee(.bright)**

**robot.whenProgramStarted**

**robot.controllerTilted( 30 )**

**robot.controllerTilted( 30 )**

**robot.whenTouched ([true, true, true, true])**

**robot.whenTouched ([true, true, true, true])**

**robot.whenControllerHearsVolume (louderThan: 5)**

**robot.whenControllerHearsVolume (louderThan: 5)**

**boomerang(orange)**

**robot.whenBoomerang( .orange )**

**robot.lockEvent { robot.move( 16 ) robot.turn( right , 90 ) }**

**if myNumber > 10 { robot.move( 10 ) }**

**if myNumber > 10 { robot.move( 10 ) }**

**else if myNumber > 5 { robot.move( 8 ) }**

**else if myNumber > 5 { robot.move( 8 ) }**

**robot.whenProgramStarted { while myNumber == 1 { robot.speeds(left: 10 , right: 10 ) }**

**! true**

**! true**

**and**

**or**

**xor**

**if myNumber < 1 { 1 < 1 }**

**if myNumber >= 1 { 1 >= 1 }**

**if myNumber > 1 { 1 > 1 }**

**myLogic = true**

**myLogic**

**1 + 1**

**myNumber + 1**

**myNumber - 1**

**myNumber \* 1**

**myNumber ^ 1**

**1 % myNumber**

**myNumber ^ 1**

**myNumber / 1**

**sqrt( 1 )**

**myNumber**

**random(integer, from: 0, to: 100)**

**random(integer, from: 0, to: 100)**

**myNumber = 0**

**myNumber = 0**

**myNumber + 1**

**myNumber**

**robot.lights( .on , Color(red)**

**robot.markerDown()**

**robot.markerDown()**

**robot.setSoundOutput(to: .robot)**

**robot.speeds(left: 10 , right: 10)**

**robot.speeds(left: 10 , right: 10)**

## move()

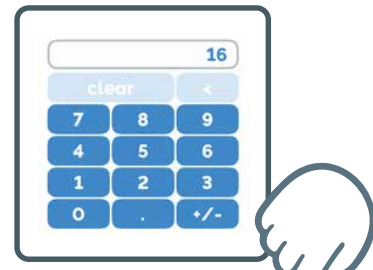
Funkce pohybu umožňuje Rootu pohybovat se vpřed nebo vzad v centimetrech.



```
robot.move( 16 )
```

## Centimeters

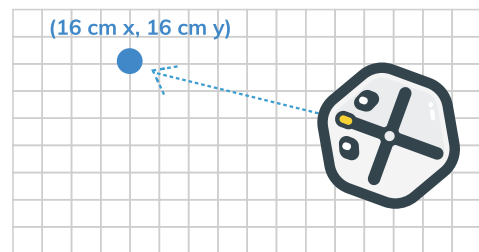
Zadejte počet centimetrů, které má Root urazit vpřed nebo vzad.



```
robot.move( 16 )
```

## navigateTo()

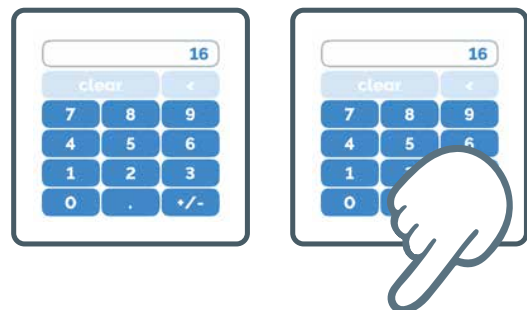
Root se může pohybovat po neviditelné čtvercové síti podle souřadnic os x a y – startovní pozice Roota je 0 cm, 0 cm.



```
robot.navigateTo( x:16 , y: 16)
```

## navigateTo()

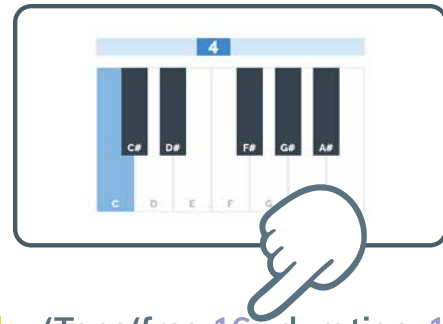
Navigujte Roota na specifickou pozici zadáním souřadnic do editoru.



```
robot.navigateTo( x:16 , y: 16)
```

## play()

Díky tomuto příkazu můžete nechat Roota hrát různé noty. Otevřete první editor pro výběr frekvence.



```
play(Tone(freq:16 , duration: 16))
```

## Duration

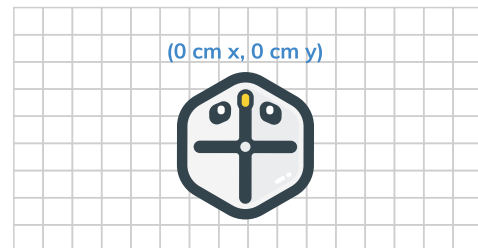
Otevřete druhý editor a určete, kolik vteřin má Root hrát vybranou notu.



```
play(Tone(freq:16 , duration: 16))
```

## resetNavigation()

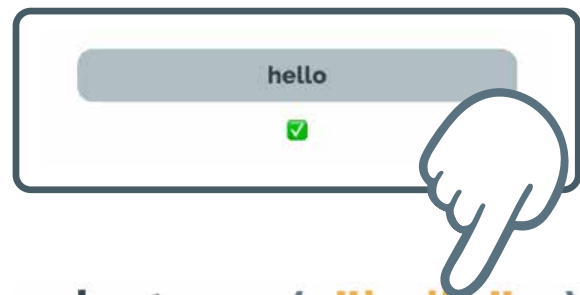
Tento příkaz nastaví aktuální pozici Roota na neviditelném čtvercovém poli jako pozici startovní (0 cm, 0 cm).



```
robot.resetNavigation()
```

## say()

Příkaz, aby Root nebo vaše zařízení "řeklo" něco nahlas.

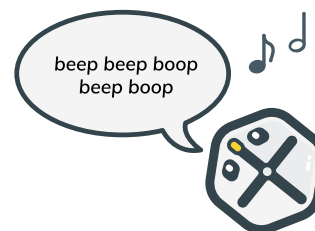


```
robot.say( "hello" )
```

## Root Language

Pokud budete kódovat Roota, aby mluvil, Root si slova přeloží do svého jazyka.

```
robot.say( "hello" )  
robot.setSoundOutput(to: .robot )
```



## turn()

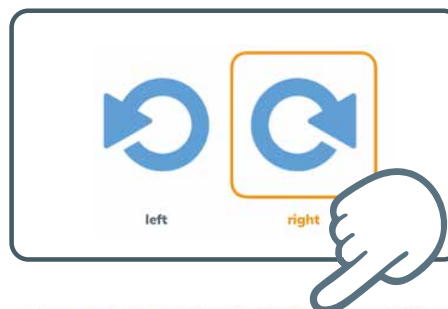
Příkaz pro otáčení Roota vpravo nebo vlevo.



```
robot.turn( right , 90 )
```

## Direction

V prvním editoru nastavte jakým směrem se má Root otočit.



```
robot.turn( right , 90 )
```

## Degrees

V druhém editoru nastavte o kolik stupňů se má Root otočit.



```
robot.turn( right , 90 )
```

## arc()

Tento blok řídí Roota podle tvaru kruhu.

```
robot.arc( right , angle: 90 , radius: 12 )
```

### arc() angle

Úhel v prvním poli říká Rootu, jak daleko má jet po kružnici.

```
robot.arc( right , angle: 90 , radius: 12 )
```



### arc() radius

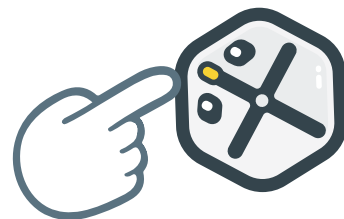
Rádus bloku oblouk říká Rootu, jaký je poloměr kružnice, po které má jet.

```
robot.arc( right , angle: 90 , radius: 12 )
```



## whenBumperPressed()

Přední nárazníky Roota mohou být nastaveny jako výchozí akce, která spouští další program.

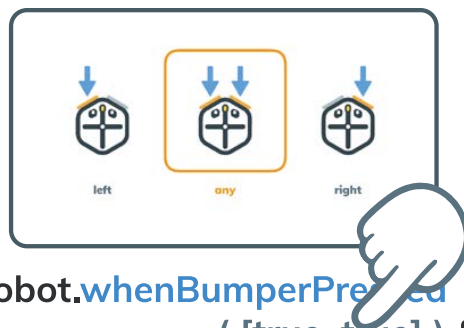


```
robot.whenBumperPressed  
( [true, true] ) {
```



## Bump Editor

Nastavte, na který z nárazníků má Root reagovat.



```
robot.whenBumperPressed  
( [true, true] ) {
```

## whenColorScanned()

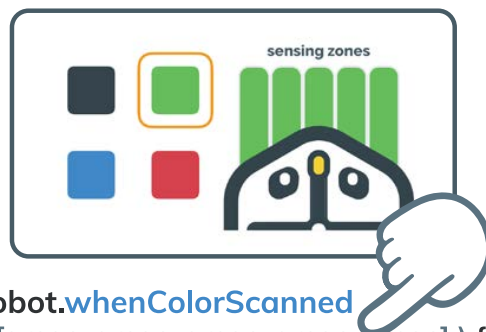
Tento blok použijte pro detekci barev a reakci Roota na barvu, kterou zvolíte.



```
robot.whenColorScanned  
( [color, color, color, color, color] ) {
```

## Color Sensor Editor

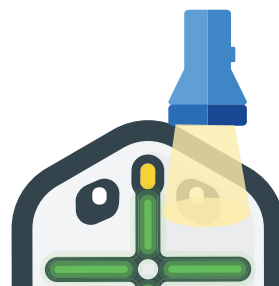
V editoru nastavte barvy a zóny, které má Root číst.



```
robot.whenColorScanned  
( [green, green, green, green, green] ) {
```

## whenEyesSaw()

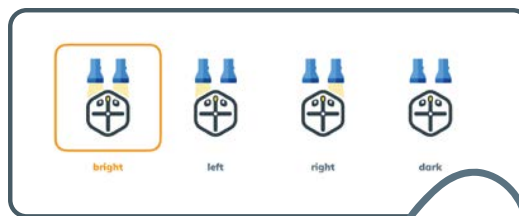
Prostřednictvím zařízení s iOS má Root možnost reagovat na změny světla.



```
robot.whenEyesSee( .bright ) {
```

## Light Editor

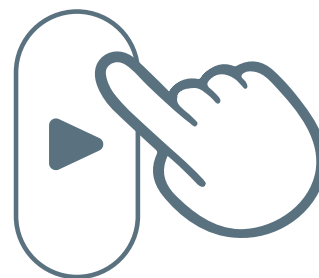
Root má dva světelné senzory, které mohou reagovat na změnu intenzity světla.



```
robot.whenEyesSee( .bright ) {
```

## whenProgramStarted()

Kód umístěný za tento příkaz se spustí, jakmile kliknete na tlačítko start.



```
robot.whenProgramStarted {
```

## whenControllerTilted()

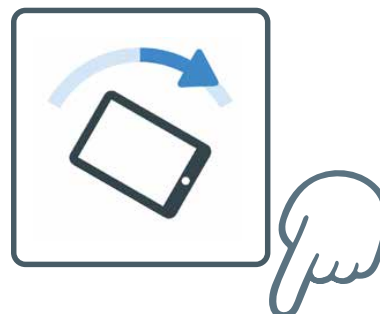
Příkaz umožňuje řídit Roota pomocí náklonu zařízení.



```
robot.controllerTilted( 30 ) {
```

## Tilt Editor

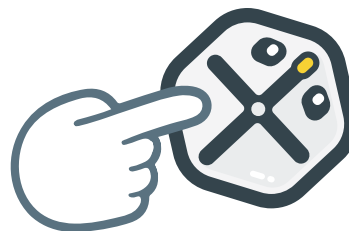
Přetáhněte šipku pro určení úhlu náklonu, podle kterého bude Root reagovat.



```
robot.controllerTilted( 30 ) {
```

## whenTouched()

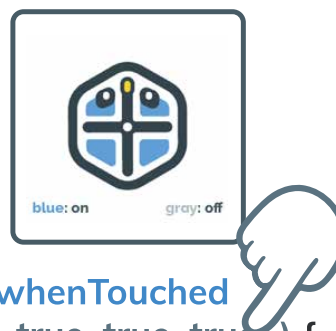
Nastavte Roota, aby reagoval na stisknutí jedné ze čtyř horních dotykových ploch.



```
robot.whenTouched  
( [true, true, true, true] ) {
```

## Touch Editor

V editoru můžete určit, které dotykové plochy reagují na dotyk, a které ne.



```
robot.whenTouched  
( [true, true, true, true] ) {
```

## whenControllerHeardVolume()

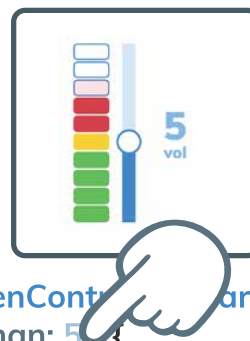
Root umí prostřednictvím zařízení s iOS rozpoznávat a reagovat na změny v hlasitosti.



```
robot.whenControllerHearsVolume  
( louderThan: 5 ) {
```

## Sound Editor

Nastavte úroveň hlasitosti, na kterou má Root reagovat.



```
robot.whenControllerHearsVolume  
( louderThan: 5 ) {
```

## Conditional Statements

Podmínky říkají Rootu, co má udělat, POKUD (if) je nějaká akce PRAVDA (true) nebo NEPRAVDA (false).



```
if random (integer, from: 0, to: 100) > 5 {  
  robot.lights( .on, Color(red: 100, green: 45, blue: 0))  
}  
else {  
  robot.lights( .on, Color(red: 0, green: 0, blue: 100))  
}
```

### if true { }

Je-li výrok uvnitř PRAVDA, program provede obsažené příkazy. Pokud je výrok NEPRAVDA, program pokračuje na další výrok.

```
if myNumber > 10 {  
  robot.move( 10 )  
}
```

### else if true { }

Pokud je výrok uvnitř bloku if true {} NEPRAVDA program pokračuje na blok else if true {}. Pokud je výrok PRAVDA, bude tento kód proveden. Pokud výrok pravdivý není program pokračuje na další blok.

```
else if myNumber > 5 {  
  robot.move( 8 )  
}
```

## else { }

Pokud je výrok uvnitř bloků if true { } a všech bloků else if true { } neplatný, program bloku else{ } je spuštěn.

```
else if myNumber > 5 {  
    robot.move( 8 )  
}  
else {  
    robot.move( 2 )  
}
```

## lockEvent { }

Když program Rootu dosáhne bloku lockEvent { }, projde celý kód, který tento blok obsahuje bez přerušení dalšími událostmi.

```
robot.whenBoomerang( .orange ) {  
    robot.lockEvent {  
        robot.move( 16 )  
        robot.turn( right , 90 )  
    }  
}
```

## for \_ in 1 ... x { }

Slouží k vytvoření smyčky kódu, která se opakuje „x“ – krát.



```
for _ in 1 ... 4 {  
    robot.move( 10 )  
    robot.turn( right, 90 )  
}
```

## wait()

Blok umožňuje vytvořit časové zpoždění před pokračováním na další blok.



wait( 1 )

## while true { }

Blok while true { } může nabývat hodnot PRAVDA nebo NEPRAVDA.



while true { }

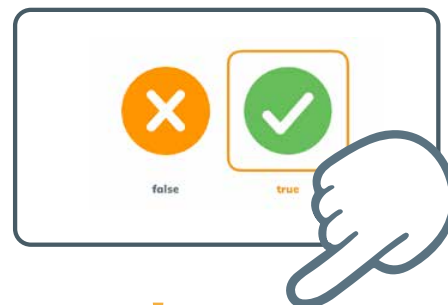
## While Loop

Pokud je výrok uvnitř tohoto bloku PRAVDA je tento kód opakován, dokud se výrok nezmění na NEPRAVDU.

```
robot.whenProgramStarted {  
  while myNumber == 1 {  
    robot.speeds(left: 10 , right: 10 )  
  }  
}
```

## !x

Otáčí výrok na jeho opak. PRAVDA se mění na NEPRAVDU a NEPRAVDA na PRAVDU.

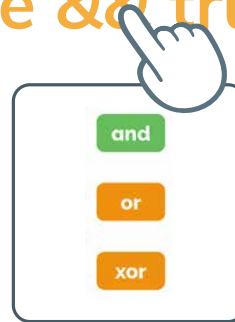


! true

## Dual Operator Block

Booleanské operátory budou buďto PRAVDA nebo NEPRAVDA v závislosti na stavu jejich parametrů.

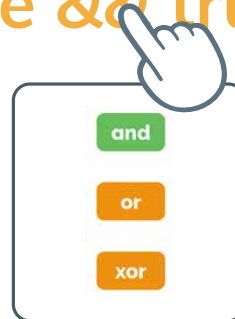
true && true



$x \ \&\& \ y$

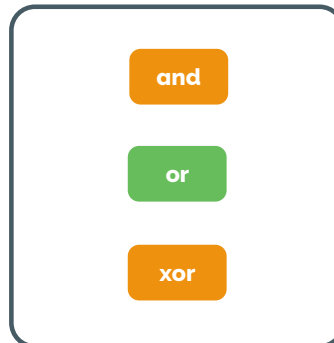
Vrací PRAVDA pokud jsou pravdivé všechny výrazy.

true && true



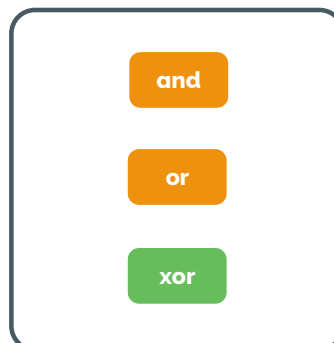
$x \ || \ y$

Vrací PRAVDA pokud je pravdivý kterýkoli z výrazů.



$x \ \wedge \ y$

Vrací PRAVDA pokud je pravdivý pouze jeden z výrazů.



## Using Conditionals

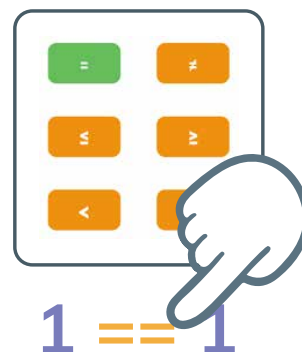
Použijte logické operace s podmínkami, a řekněte tak Rootovi, co má dělat, pokud jsou výrazy uvnitř pravdivé nebo nepravdivé.

```
if (myNumber < 1 && theirNumber > 10) {
```



## Comparison Operators

Srovnávací blok vrátí buď PRAVDA (true), nebo NEPRAVDA (false) v závislosti na vztahu mezi jednotlivými poli.



1 == 1

**x == y**

Vrací PRAVDA pokud je první hodnota stejná jako hodnota druhá.

```
if (myNumber == 1) {
```

1 == 1

**x != y**

Vrací PRAVDA pokud není první hodnota stejná jako hodnota druhá.

```
if (myNumber != 1) {
```

1 != 1



**$x \leq y$**

Vrací PRAVDA pokud je první hodnota stejná nebo nižší oproti hodnotě druhé.

```
if (myNumber <= 1) {  
    1 <= 1
```

**$x < y$**

Vrací PRAVDA pokud je první hodnota nižší než hodnota druhá.

```
if (myNumber < 1) {  
    1 < 1
```

**$x \geq y$**

Vrací PRAVDA pokud je první hodnota vyšší nebo stejná oproti hodnotě druhé.

```
if (myNumber >= 1) {  
    1 >= 1
```

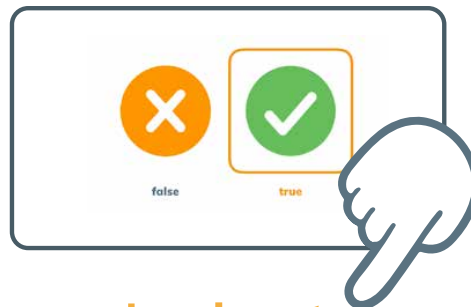
**$x > y$**

Vrací PRAVDA pokud je první hodnota vyšší než hodnota druhá.

```
if (myNumber > 1) {  
    1 > 1
```

## myLogic = x

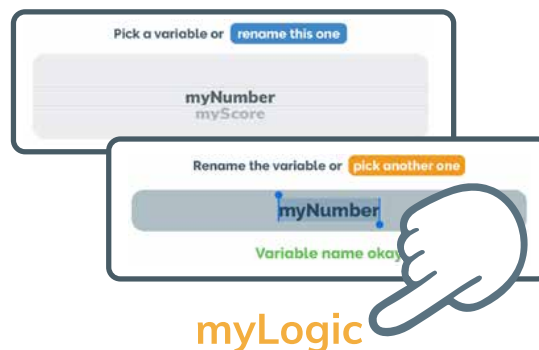
Logickou proměnnou můžete nastavit na PRAVDU (true) nebo NEPRAVDU (false) v závislosti na události, například když klepnete na nárazník.



myLogic = true

## Boolean Variable

Logické proměnné mohou přepínat mezi pravdivými nebo nepravdivými hodnotami, aby pomohly Rootovi sledovat, co se děje v projektu.

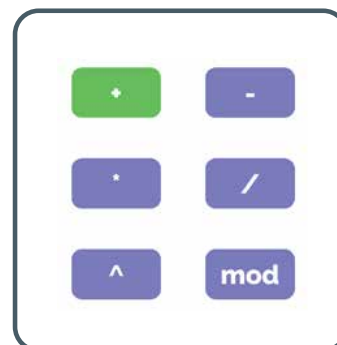


myLogic

## Math Operators

Zobrazí výsledek vybrané matematické operace.

1 + 1



$x + y$

Sčítání dvou hodnot.

myNumber + 1

**x - y**

Odčítání dvou hodnot.

**myNumber - 1**

---

**x \* y**

Násobení dvou hodnot.

**myNumber \* 1**

---

**x ^ y**

První číslo je umocněno pomocí hodnoty ve druhém poli.

**myNumber ^ 1**

---

**x % y**

Výsledek je roven zbytku po podílu první a druhé hodnoty.

**1 % myNumber ^ 1**

---

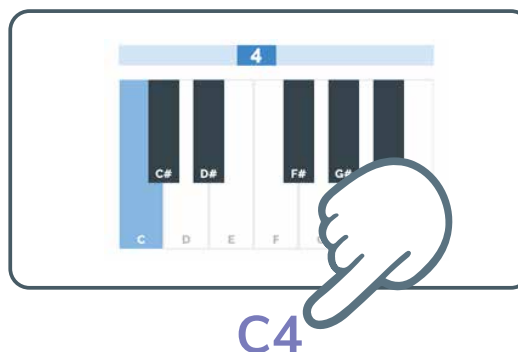
$x / y$

Dělení dvou hodnot.

myNumber / 1

## Music Frequency Block

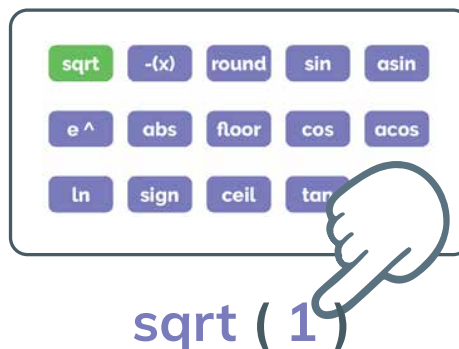
Umožňuje nastavit zvolenou frekvenci not v Hz.



C4

## Math Functions

Výsledek vybrané funkce odpovídající zadané hodnotě.



sqrt ( 1 )

## Function Editor

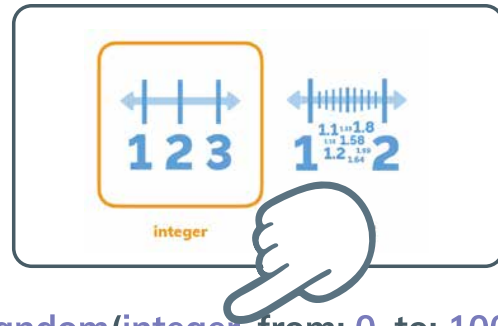
Editor umožňuje výběr požadované matematické operace.

sqrt ( 1 )  
myNumber



## random()

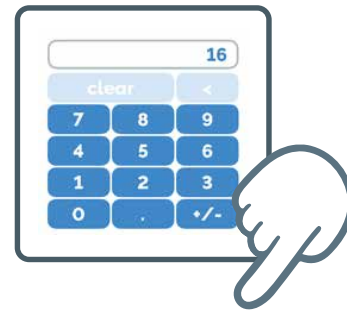
Tento příkaz generuje jakékoli celé nebo desetinné číslo uvnitř rozsahu dvou hodnot.



`random(integer, from: 0, to: 100)`

## Random Range

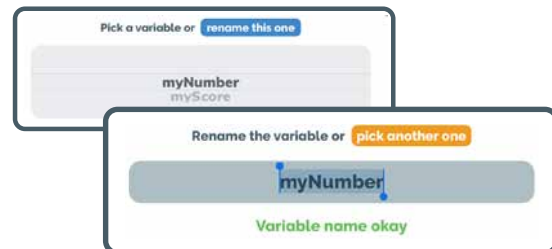
Rozsah můžete změnit pomocí dvou číselných polí.



`random(integer, from: 0, to: 100)`

## myNumber = x

Zde nastavíte počáteční hodnotu proměnné.



`myNumber = 0`

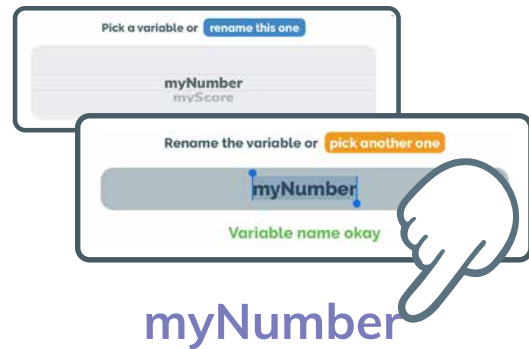
## Keeping Score

Příkaz nastavení proměnné může obsahovat číslo nebo výsledek početní operace. Lze tak například připočítat "1" k aktuální hodnotě proměnné.

`myNumber = 0`  
`myNumber + 1`

## Float Variable

Čísla proměnných udržují hodnoty, které pomáhají Rootu sledovat, co se děje v projektu.



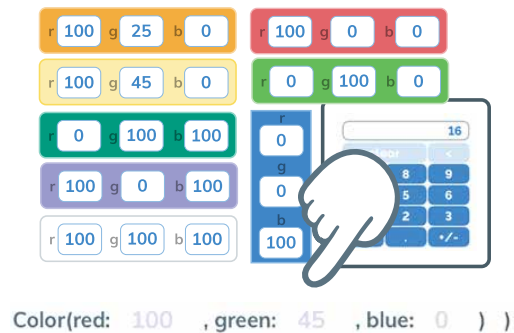
## lights()

Umožňuje rozsvítit nebo rozblíkat světlo na vrchní straně Roota, měnit jeho barvu nebo změnit styl svícení.



## Light Editor

Použijte editor ke změně barvy a stylu svícení. Všechny barvy světla jsou směsí červené, zelené a modré barvy. Vytvořte různé barvy nastavením různých hodnot R, G, B na různá čísla.



## markerDown()

Příkaz se používá pro zasunutí a vysunutí fixu a mazací plochy.



## Marker Editor

Použijte editor pro určení, má-li se Root připravit ke psaní nebo k mazání.



```
robot.markerDown ()
```

## soundOutput()

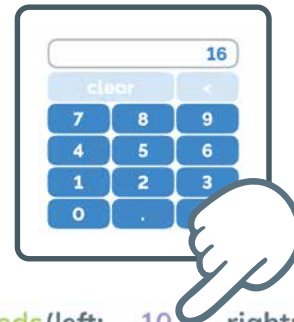
Nastavte přehrávání zvuků na vašem zařízení s iOS nebo prostřednictvím Roota.



```
robot.setSoundOutput(to: .robot )
```

## speeds()

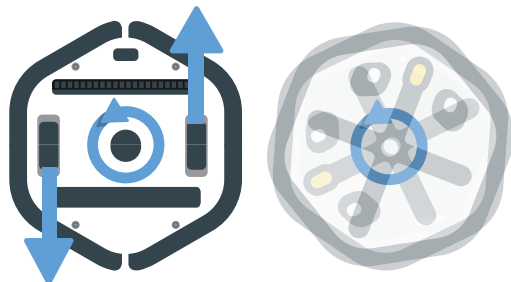
Nastavte rychlost kol pro změnu rychlosti a pro změnu poloměru otáčení Roota.



```
robot.speeds(left: 10 , right: 10 )
```

## Wheel Speeds Editor

Nastavte rychlost jednoho z kol na negativní hodnotu. Jak se bude Root pohybovat?



```
robot.speeds(left: 10 , right: 10 )
```