

Ovládnite opakovanie: Od jednoduchých príkazov k inteligentným systémom

Sprievodca cyklami v prostredí Microsoft MakeCode pre micro:bit



Prečo sú cykly srdcom každého programu?

Predstavte si, že musíte každý príkaz napísať stokrát. Cykly sú elegantným riešením. Umožňujú nám automatizovať opakujúce sa úlohy, vytvárať animácie, reagovať na vstupy a vdýchnuť život našim projektom. Sú kľúčom k efektívnemu a výkonnému kódu.



Automatizácia: Vykonajte blok kódu viackrát bez jeho kopírovania.



Interaktivita: Udržujte program v chode, kým čakáte na stlačenie tlačidla alebo zmenu senzora.



Efektivita: Vytvárajte zložité správanie s minimom kódu.

Viac práce



Inteligentná práca



Menej práce



Naša cesta k majstrovstvu

Prejdeme od základných opakovaní až po pokročilé techniky riadenia. Každá časť predstavuje nový stupeň kontroly nad vaším programom.



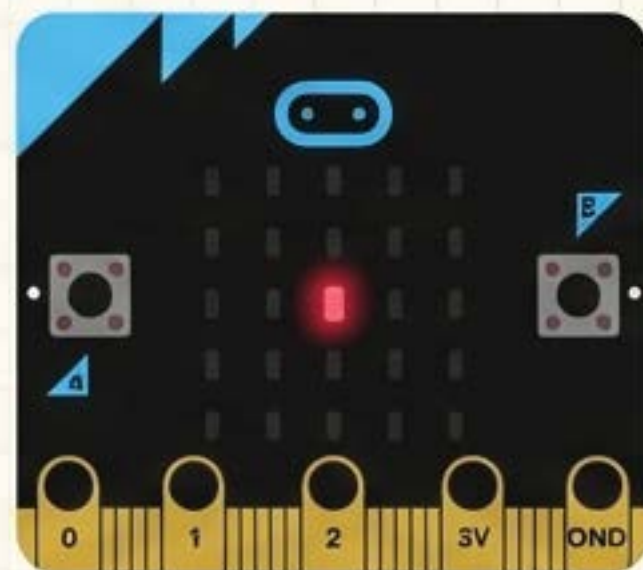
Príkaz `opakovať 4 krát`

Základná myšlienka

Pre situácie, kedy potrebujete akciu zopakovať vopred stanovený počet krát.

Ideálne pre

Jednoduché animácie, úvodné sekvencie alebo krátke melódie.



Ako to funguje?

- Nastavenie opakovania:** Príkaz určí, že nasledujúci blok kódu sa má vykonať štyrikrát.
- Vykonanie kódu:** Počet opakovaní je pevne stanovený, takže vnorený kód sa vykoná presne štyrikrát.

Časť 2: Inteligentné opakovanie

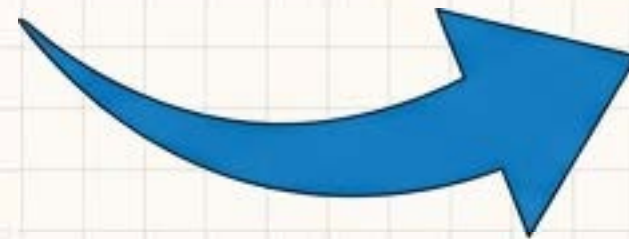
Ked' opakovanie nestačí: Potrebujeme počítat' a spracovávať

Čo ak potrebujeme v každom opakovaní urobiť niečo trochu iné? Nasledujúce cykly nám dávajú prístup k „počítadlu“ (indexu) alebo k jednotlivým prvkom v zozname. To otvára dvere k práci s dátami, súradnicami a sekvenciami.

Statické opakovanie



Dynamické opakovanie



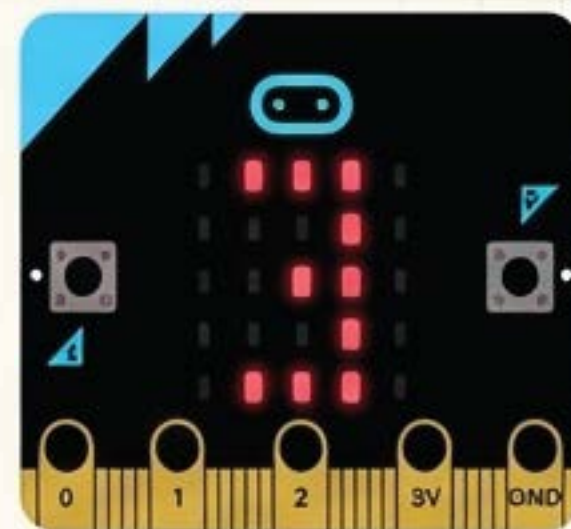
Príkaz **pre index** od 0 do 9

Základná myšlienka

Opakuje blok kódu a zároveň poskytuje premennú (**index**), ktorá sa pri každom prechode mení.

Ideálne pre

Postupné rozsvetovanie LED diód, prácu so súradnicami, matematické sekvencie alebo odpočítavanie.



Definuje začiatok a koniec počítania.

Hodnota **index** sa mení v každom kole (0, 1, 2...).

Ako to funguje?

1. **Nastavenie rozsahu:** Príkaz definuje rozsah premennej **index**, v tomto prípade od 0 do 9.
2. **Vykonanie kódu:** Blok kódu sa vykoná 10-krát. V každom opakovaní sa hodnota premennej **index** postupne zvyšuje.

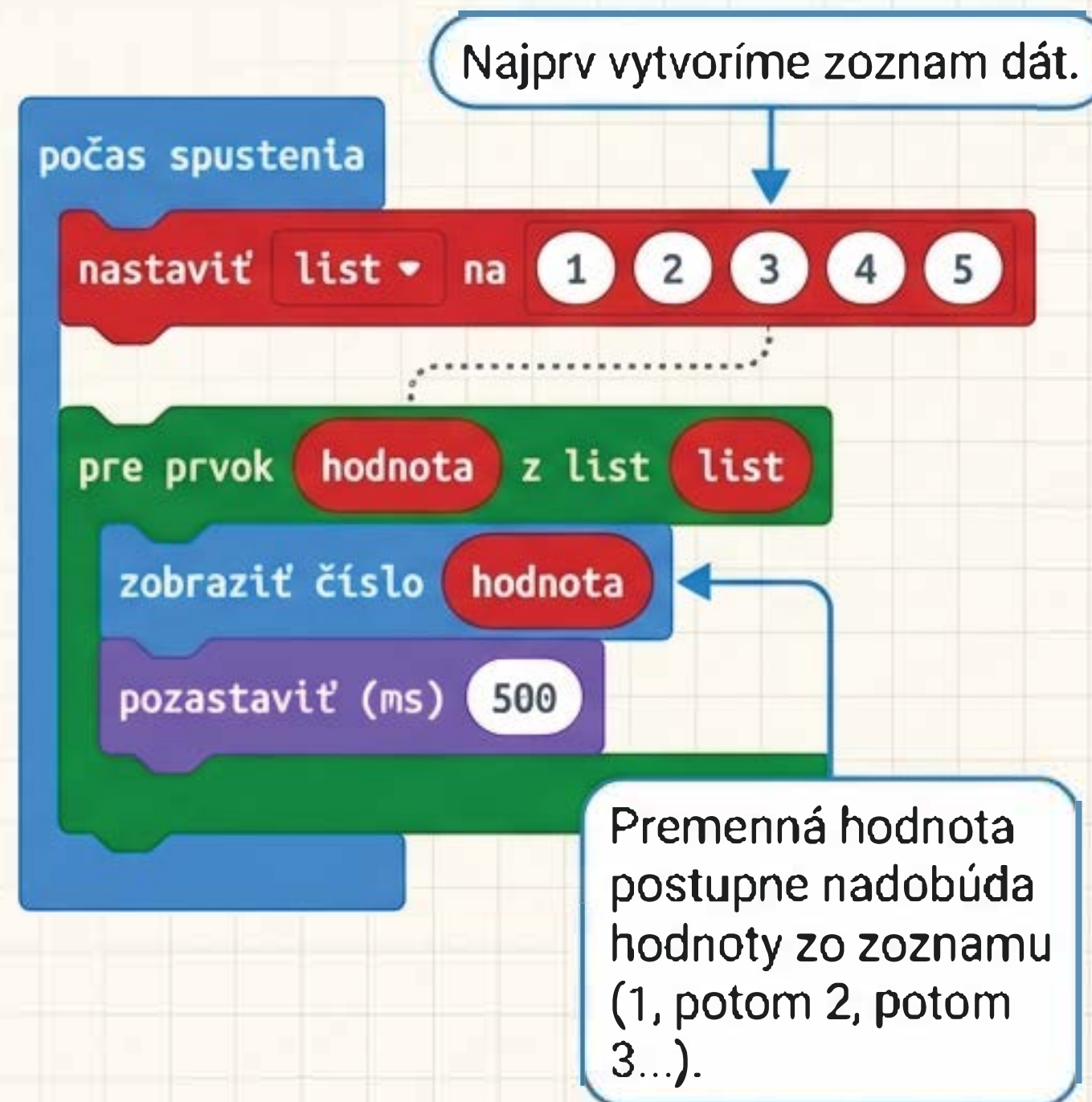
Príkaz `pre prvok hodnota z list`

Základná myšlienka

Prechádza každou položkou v zozname (liste) a umožňuje vám s ňou pracovať jednotlivo.

Ideálne pre

Zobrazovanie hodnôt zo senzorov, prehrávanie melódií zo zoznamu nôt, spracovanie textu.



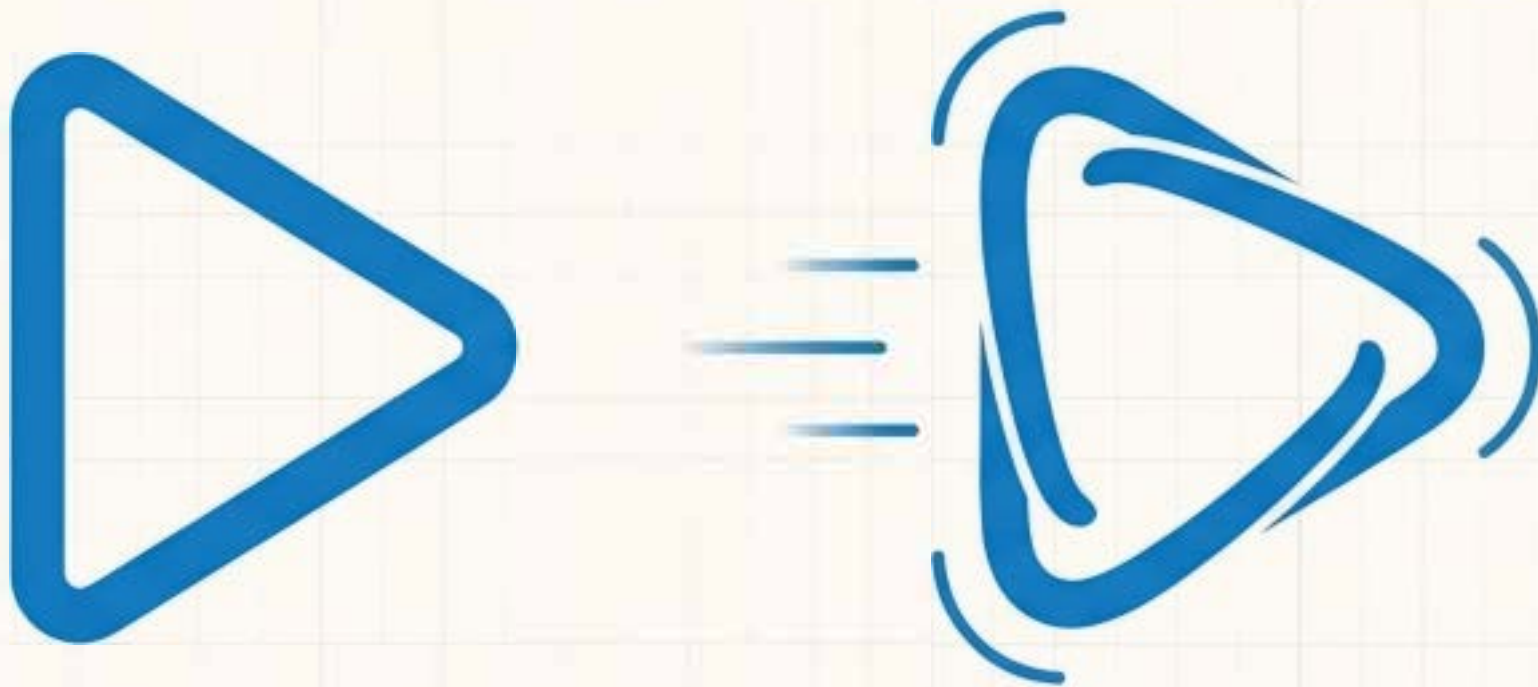
Ako to funguje?

- Nastavenie iterácie:** Príkaz definuje iteráciu cez každý prvok v určenom zozname.
- Vykonanie kódu:** Blok kódu sa vykoná pre každý prvok, pričom premenná hodnota vždy predstavuje aktuálny prvok zo zoznamu.

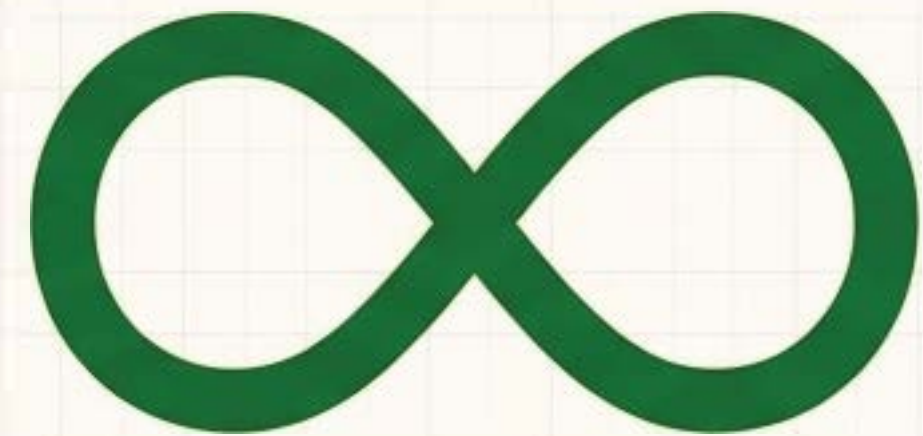
Časť 3: Nekonečné programy

Od jednorazových úloh k programom, ktoré žijú

Ako prinútiť micro:bit, aby neustále niečo robil? Aby čakal na náš pokyn alebo aby každú sekundu skontroloval teplotu? Na to slúžia cykly, ktoré bežia buď donekonečna, alebo kým platí určitá podmienka. Tieto príkazy sú základom pre interaktívne projekty.



Spustenie



Neustála aktivita

Príkaz `zatiaľ čo` [podmienka]

Základná myšlienka

Opakuje kód, kým je zadaná podmienka pravdivá. Je to cyklus, ktorý „čaká“ na zmenu stavu.

Ideálne pre

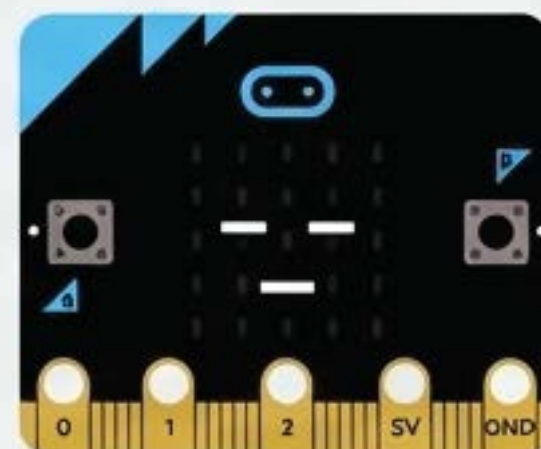
Čakanie na vstup od používateľa, reakcie na senzory (napr. „kým je teplota < 20°C“), vytváranie herných stavov.



Ako to funguje?

1. **Overenie podmienky:** Na začiatku každého opakovania sa skontroluje, či je podmienka splnená.
2. **Vykonanie alebo ukončenie:** Ak je podmienka pravdivá, kód vo vnútri sa vykoná. Akonáhle je nepravdivá, cyklus sa preskočí a program pokračuje ďalej.

Stav 1



Podmienka je pravdivá, cyklus beží.

Stav 2



Podmienka sa stala nepravdivou, cyklus sa ukončil.

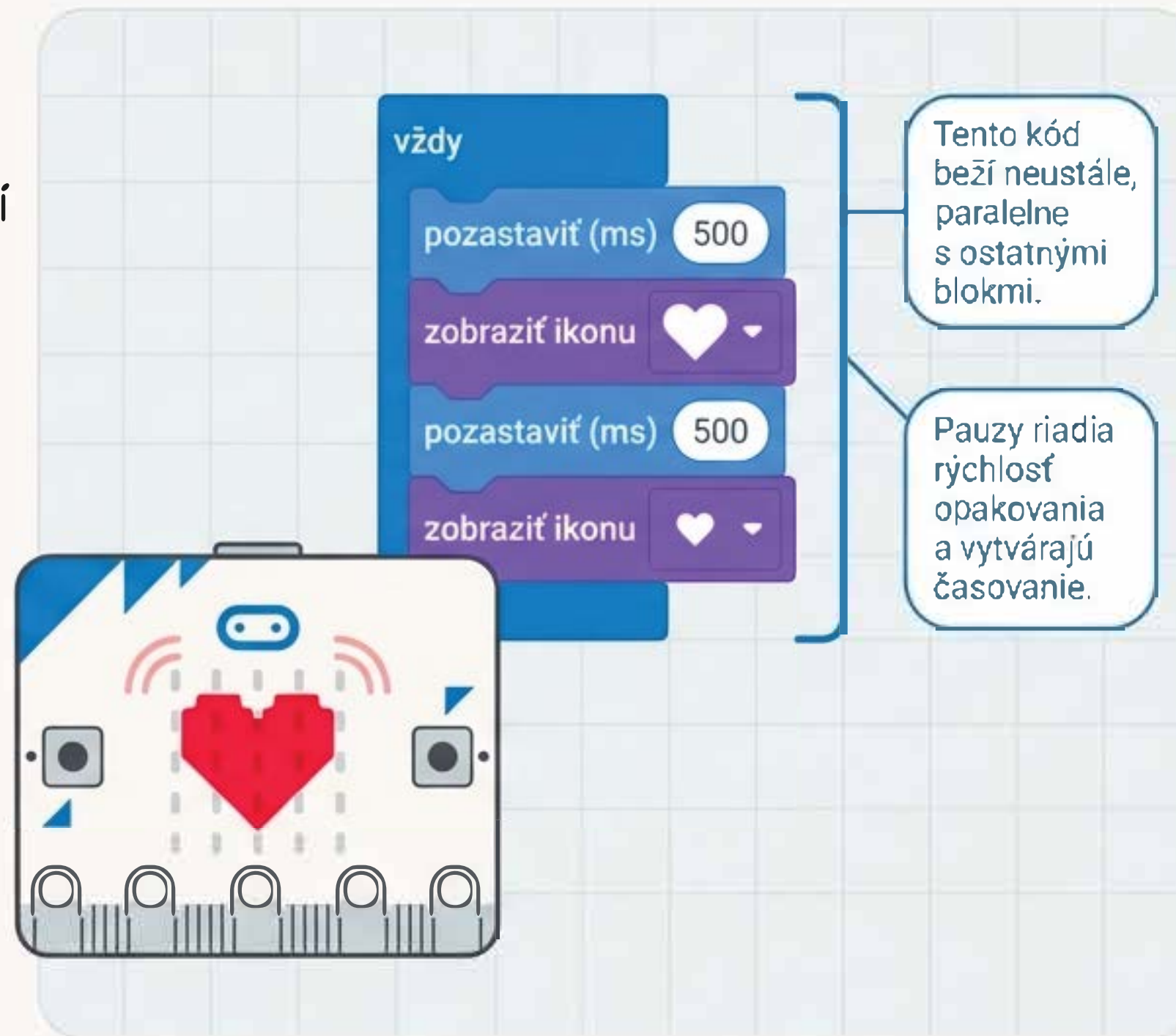
Príkaz `vždy`

Základná myšlienka

Vytvára nezávislý, nekonečne sa opakujúci proces, ktorý beží na pozadí od spustenia micro:bitu.

Ideálne pre

Animácie na pozadí,
pravidelné meranie senzorov,
„srdcový tep“ projektu.



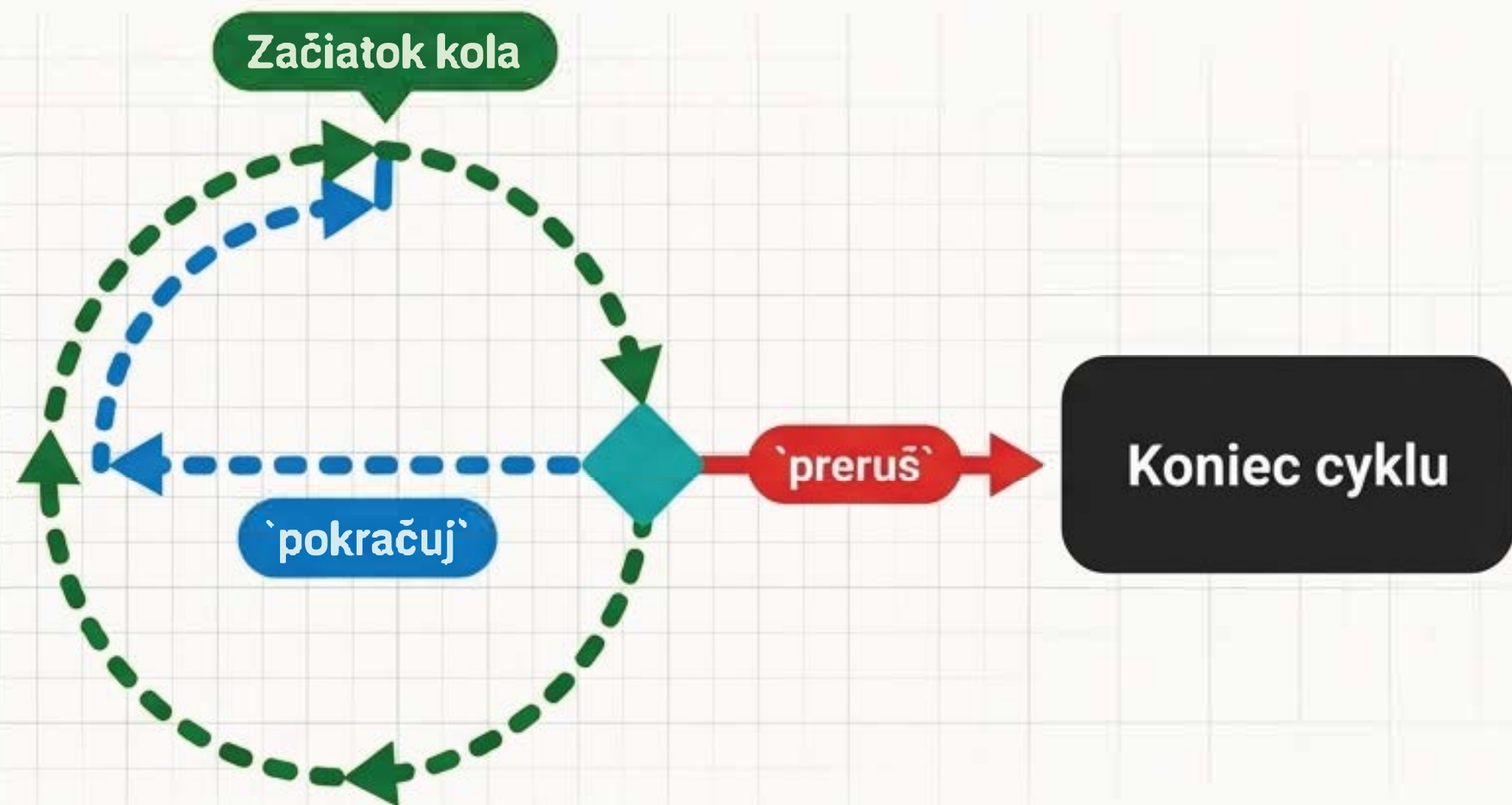
Ako to funguje?

1. **Nekonečné opakovanie:** Kód v tomto bloku sa opakuje neustále, od spustenia micro:bitu až do jeho vypnutia alebo resetu.
2. **Paralelný beh:** Beží súbežne s blokmi ako *`počas spustenia`* alebo *`keď je tlačidlo A stlačené`*, čo umožňuje vykonávať viac úloh naraz.

Časť 4: Precízna kontrola

Ked' potrebujete porušiť pravidlá

Cykly majú svoj predvídateľný priebeh. Ale čo ak potrebujeme cyklus opustiť predčasne alebo len preskočiť jedno konkrétne kolo? Príkazy ``preruš`` a ``pokračuj`` nám dávajú túto chirurgickú presnosť a umožňujú nám riadiť tok programu do najmenších detailov.



Príkaz `preruš`

Základná myšlienka

Okamžite a úplne ukončí vykonávanie aktuálneho cyklu.

Ideálne pre

Hľadanie prvej zhody v dátach, ukončenie hry pri splnení víťaznej podmienky, reakciu na kritický signál.



Ako to funguje?

1. ****Detekcia príkazu*:** Keď program vo vnútri cyklu narazí na príkaz `preruš`, okamžite prestane vykonávať ďalšie príkazy v cykle.
2. ****Ukončenie slučky*:** Vykonávanie programu preskočí za celý blok cyklu a pokračuje ďalšími príkazmi.

Príkaz `pokračuj`

Základná myšlienka

Preskočí zvyšok aktuálneho kola cyklu a prejde rovno na ďalšiu iteráciu.

Ideálne pre

Filtrovanie dát (ignorovanie nechcených hodnôt), vytváranie výnimiek v správaní, optimalizáciu.



Očakávaný výstup:

0, 1, 2, 3, 4, [pauza], 6, 7, 8, 9

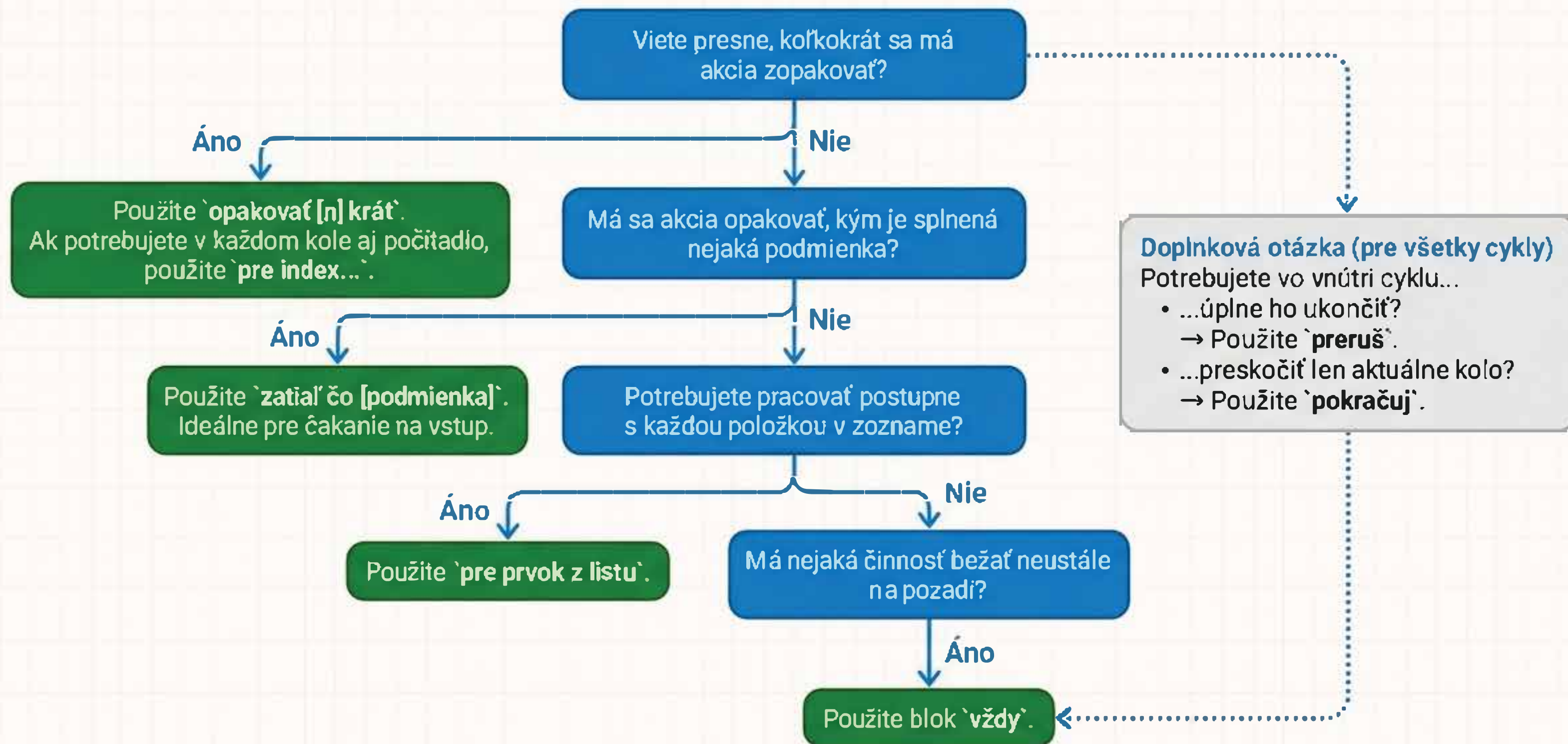
Číslo 5 bolo preskočené.

Ako to funguje?

1. ****Detekcia príkazu****: Keď program narazí na `pokračuj`, okamžite preskočí všetky zvyšné príkazy v aktuálnom opakovaní.
2. ****Prechod na ďalšie opakovanie****: Vykonávanie sa presunie na začiatok cyklu a spustí ďalšiu iteráciu (napr. s ďalšou hodnotou `index`).

Spojenie síl: Kedy ktorý cyklus použiť?

Výber správneho cyklu je kľúčom k čistému a funkčnému kódu. Tento sprievodca vám pomôže rozhodnúť sa.



Od príkazov k tvorivosti

Cykly nie sú len technické príkazy. Sú to základné stavebné kamene pre dynamiku, interaktivitu a inteligenciu vašich projektov. Teraz máte nástroje na to, aby ste premenili jednoduché sekvencie na systémy, ktoré žijú, reagujú a tvoria.

Experimentujte. Kombinujte. Vytvorte niečo úžasné.

