



IRIM - Institut za razvoj
i inovativnost mladih

micro:bit u nastavi

Zbrajanje i oduzimanje cijelih brojeva od -9 do 9

6. razred



MATEMATIKA

○ Zbrajanje i oduzimanje cijelih brojeva od -9 do 9

Izradite program pomoću kojeg možete vježbati zbrajanje i oduzimanje cijelih brojeva, kad su zadani brojevi mali, od -9 do 9. Pritiskom lijeve i desne tipke zajedno, u pozadini se (bez prikazivanja na ekranu) generira novi zadatak, odnosno dva broja između -9 i 9 koje će trebati zbrojiti ili oduzeti. Pritom se na ekranu prikaže smješko. Pritiskom na lijevu tipku, ispisuje se zadatak, npr. -9-6. Učenik tada treba razmisliti koje je rješenje. Pritiskom na desnu tipku, rješenje se ispisuje. Prije nego se ponovo stisnu obje tipke istovremeno, svaka se tipka posebno može stisnuti koliko se puta želi. Pritom lijeva tipka uvijek daje (isti) zadatak, a desna tipka rješenje. Tek pritiskom obje tipke zajedno, u pozadini se generira novi zadatak, a na ekranu pojavljuje smješko.

Ovu lekciju možete pronaći na edukacijskom portalu Izradi! na poveznici:

<https://bit.ly/zbrajanje-i-oduzimanje-9-9>.

○ Korištenje s učenicima

Predmet	Matematika
Razred	6., osnovna škola
Polugodište	2.
Nastavna cjelina	Cijeli brojevi
Tema	Zbrajanje i oduzimanje cijelih brojeva od -9 do 9, automatizacija, račun napamet
Ključne riječi	cijeli brojevi, zbrajanje, oduzimanje, automatizacija, račun napamet
Ideja i izrada materijala	Antonija Horvatek

Izradite program pomoću kojeg učenici mogu vježbati na razne načine. Jedan je da vježbaju samostalno kod kuće, na način prikazan u videu. Na isti način mogu vježbati i na dopunskoj nastavi, na kojoj obično imate učenike s različitim teškoćama/potrebama, zbog čega za svakoga posebno treba pripremiti materijale. Učenici koji imaju poteškoća s ovakvim računom, jednostavno i samostalno mogu vježbati pomoći micro:bita. Osim što vježbaju gradivo matematike, time ih ujedno navikavate na samoprocjenu svog znanja (kroz povratnu informaciju koju micro:bit daje o točnosti rezultata) i na uočavanje da i samostalnim radom (bez učitelja) mogu napredovati. Učenicima koji tako vježbaju, treba preporučiti da, kad dođu do zadatka čije im rješenje nije jasno, pozovu učitelja. Dok učitelj s ostalim učenicima nešto radi, ovaj učenik samostalno vježba s micro:bitom, a kad učitelj dođe do tog učenika, može ga ispitati bez micro:bita.

Na redovnoj nastavi također možete koristiti micro:bit s danom datotekom. Jedan je način da na kraju sata, na kojem ste zajedno vježbali automatizaciju ovakvih zadataka, napravite igru/natjecanje između učenika. Učenici će igrati u parovima. Na papiru trebaju napraviti tablicu s dva retka, na početku redaka napisati svoja imena, a zatim naizmjenice rješavati zadatke koje

zadaje micro:bit, te u tablicu kod svog imena napisati plus ako je točno riješen zadatak, odnosno minus ako je netočno. Na donjoj slici prikazan je primjer u kojem je Ivan prvi zadatak točno riješio, Marko svoj prvi zadatak netočno, zatim je Ivan drugi zadatak riješio točno, Marko svoj također točno itd. Naravno, zadaci se (zbog dinamike i veće zanimljivosti) rješavaju usmeno – učenik protivnik će paziti je li izgovoreno rješenje točno.

Opis programa

Izradite program pomoću kojeg možete vježbati zbrajanje i oduzimanje cijelih brojeva (od -9 do 9).

Usput naučite: definiranje varijable koja poprima slučajnu vrijednost između -9 i 9 tako da češće bude odabrana negativna vrijednost (posebno će se odabirati predznak, a posebno apsolutna vrijednost), definiranje znakovne varijable čiji će zapis biti jednak vrijednosti brojevnog varijable (npr. $7 \rightarrow "7"$), ispis mješovitog zapisa sastavljenog iz brojevnih i znakovnih varijabli, korištenje ugniježđenih `if then esle` naredbi (jedne unutar drugih).

Program treba raditi tako da:

Pritiskom tipke A i tipke B istovremeno, ispiše se smješko i slučajnim odabirom se izabiru dva broja između -9 i 9 (ali se tad ne ispisuju). Pritom se negativni brojevi trebaju izabirati češće od pozitivnih.

Pritiskom tipke A ispisuje se zadatak, npr. $-8 - 9$.

Pritiskom tipke B ispisuje se rezultat.

Program se izrađuje u MakeCode editoru kojem možete pristupiti preko poveznice:

<https://makecode.microbit.org/>.

Izrada programa

1. korak

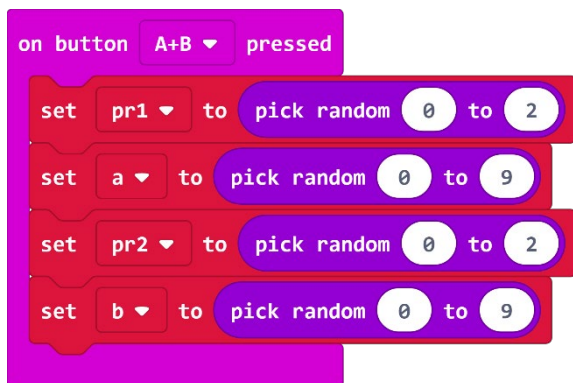
Iz skupine **Input** izaberite naredbu **on button A pressed**, kliknite na strelicu pokraj A i izaberite A+B. Sve što ćete ugnijezditi unutar ove naredbe, izvršit će se kad na micro:bitu istovremeno stisnete tipke A i B.



2. korak

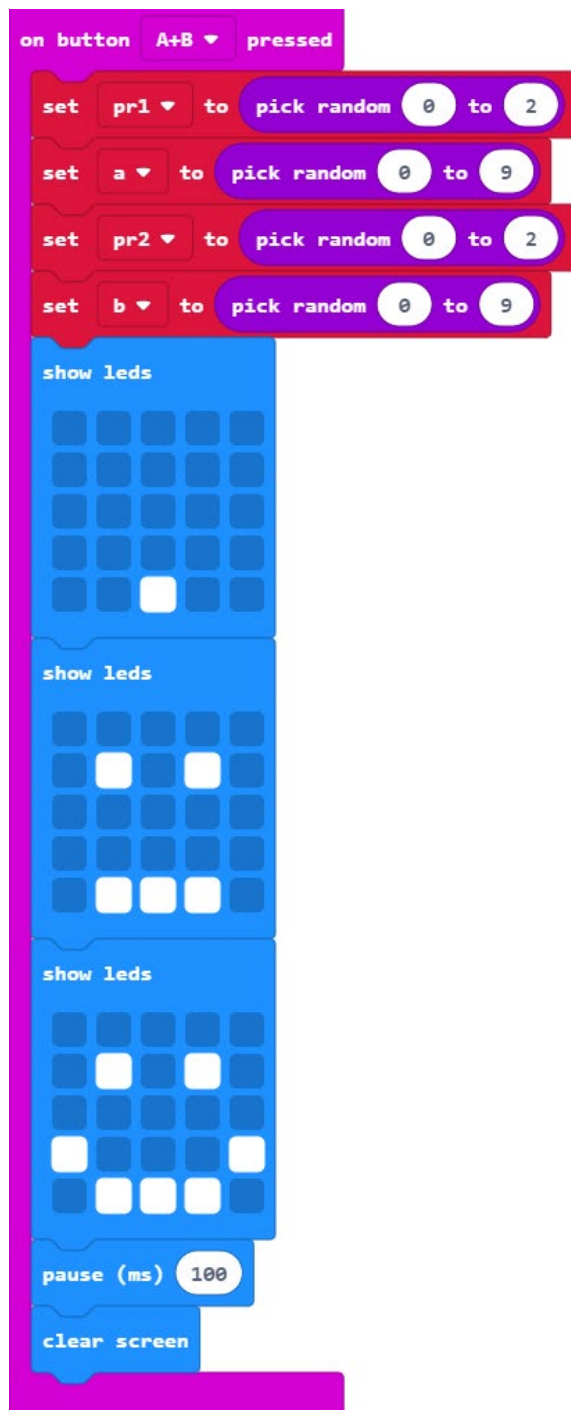
Budući da ovdje želite da se češće pojavljuju negativni brojevi nego pozitivni, naredbu za slučajni odabir broja (**pick random**) nećete koristiti na način da se jednostavno izabere broj između -9 i 9, već ćete posebno izabirati predznak (tako da se minus izabire češće nego plus), a posebno apsolutnu vrijednost od 0 do 9. Tako ćete učiniti za oba broja koja želite generirati.

Definirate varijable **pr1**, **a**, **pr2** i **b**. Varijabla **pr1** predstavlja predznak, dok varijabla **a** predstavlja apsolutnu vrijednost prvog broja. Varijable **pr2** i **b** predstavljaju predznak i apsolutnu vrijednost drugog broja. Varijable **pr1** i **pr2** mogu slučajnim odabirom poprimiti vrijednosti 0, 1 i 2. Ako poprime vrijednost 0, uzet ćete da je predznak plus, a ako poprime vrijednosti 1 ili 2, to će predstavljati predznak minus. Budući da više brojeva predstavlja minus, time će biti osigurano da se češće pojavljuju negativni brojevi nego pozitivni (dvostruko češće).



3. korak

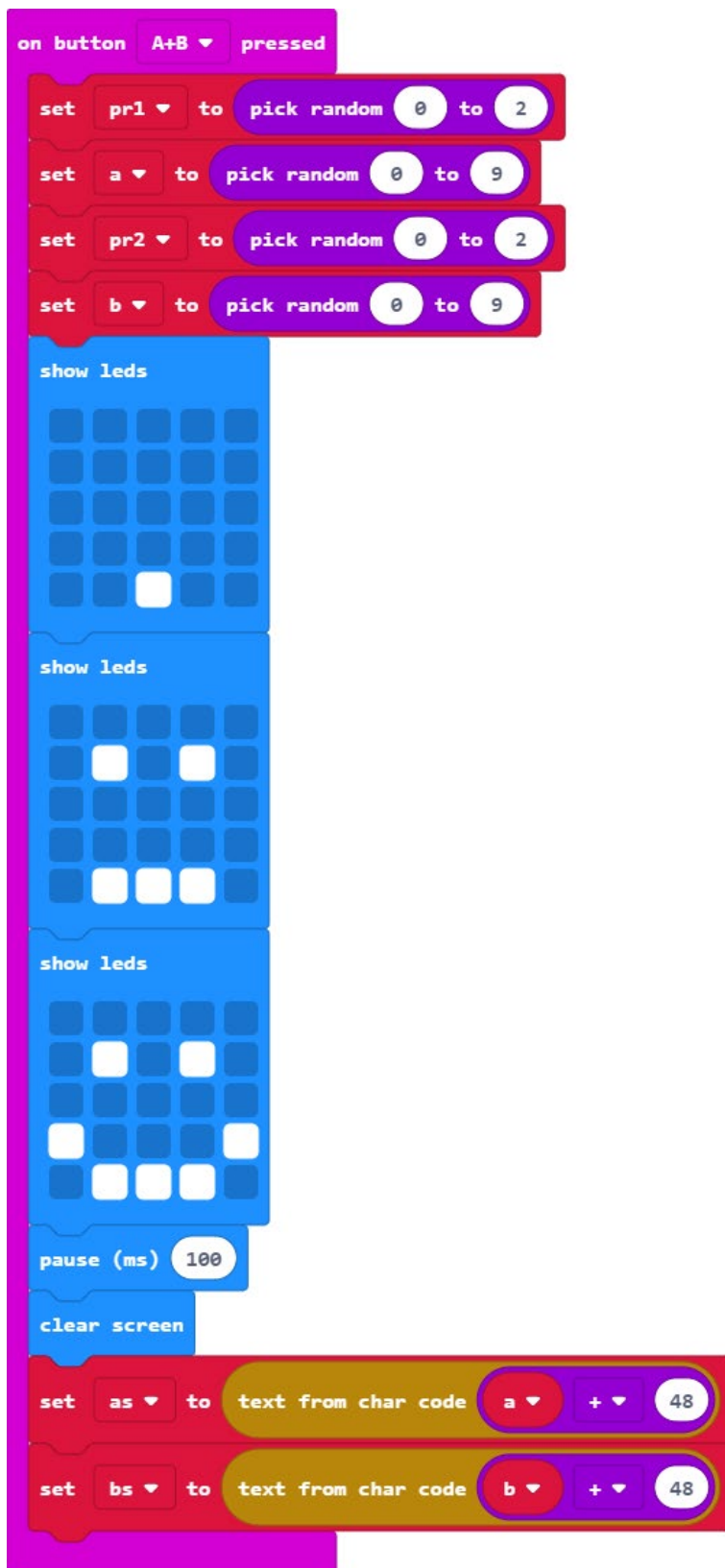
Sad ćete unutar bloka **on button A+B pressed** dodati i naredbe za iscrtavanje smješka (**show leds**), a nakon toga i za pauzu (**pause (ms) 100**) da se smješko kratko zadrži, te nakon toga obriše ekran (**clear screen**). Naravno, iscrtavanje smješka može biti realizirano na razne načine; meni se svidio ovaj.



4. korak

Još ćete, unutar bloka **on button A+B pressed**, dodati i definicije varijabli **as** i **bs** (što su kratice za “a string” i “b string”) koje će biti znakovi “stringovi” koji sadrže isti zapis kao i varijabla **a** odnosno **b**. Npr. ako varijabla **a** poprimi vrijednost 7, varijabla **as** će postati string varijabla “7”. Varijable **as** i **bs** će vam trebati kasnije, kod ispisa zadatka.

U svrhu definicije takvih varijabli koristit ćete ASCII tablicu, u kojoj vidite da se npr. zapis “1” nalazi pod brojem 49, zapis “2” pod brojem 50 itd., otkuda zaključujete da varijabli/broju treba dodati 48 i očitati znak s tog mjesta. U tu svrhu koristite naredbe **text from char code** (iz kategorije **Text**), kao što se vidi na slici.



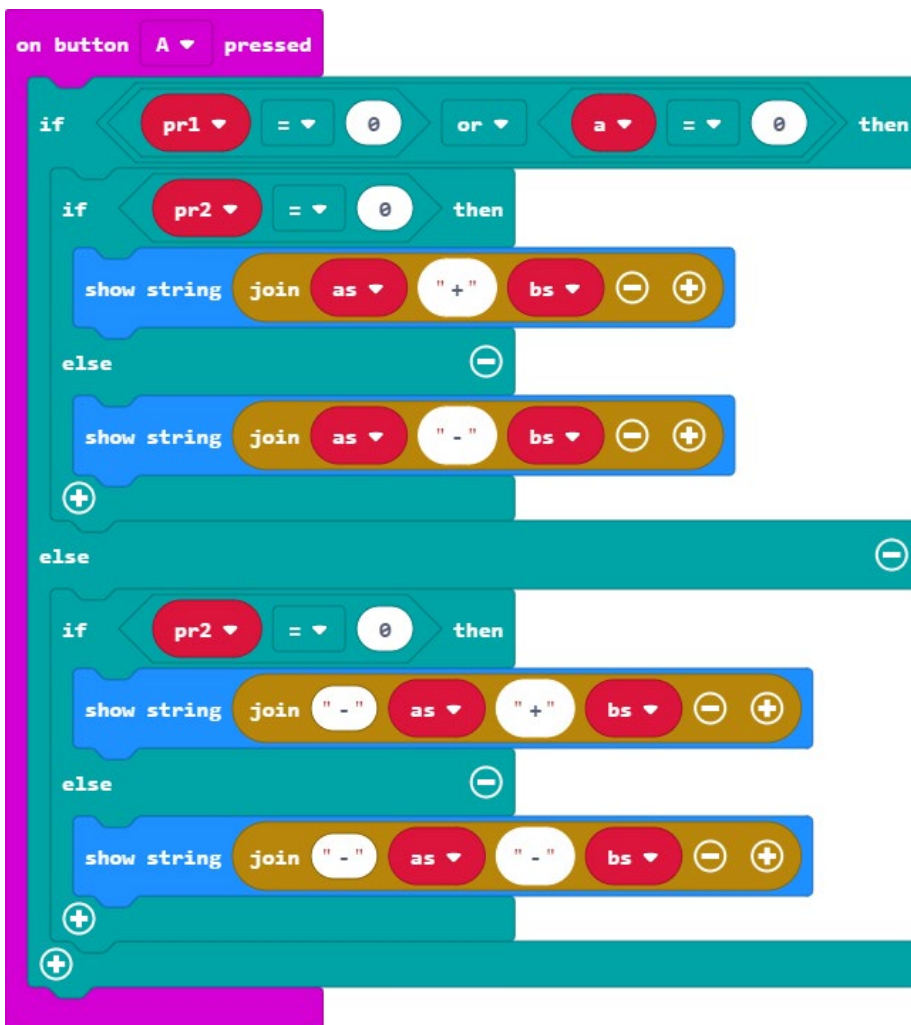
Time ste završili kodiranje dijela koji se odnosi na istovremeni pritisak na tipke A i B. Sad namjestite da se pritiskom na tipku A ispiše zadatak, npr. $-3+4$ ili $2-9$.

5. korak

Ovdje ćete morati koristiti **if then else** naredbe, budući da varijable **pr1** i **pr2** poprimaju vrijednosti brojeva od 0 do 2, a u ovisnosti o broju ispisuje se plus ili minus. Uz to, ako je prvi predznak plus, njega ne treba niti zapisati (npr. nećete ispisati +2-9, već 2-9). Isto tako, ako je prva apsolutna vrijednost 0, također nećete pisati nikakav predznak ispred, bez obzira na vrijednost varijable **pr1** (npr. želite ispis 0-6, a ne -0-6).

Prva **if** naredba provjerava je li **pr1=0** (što predstavlja početni predznak plus) ili je **a=0**. U oba ta slučaja ispred prve varijable ne ispisuje se nikakav predznak, već se kreće s ispisivanjem vrijednosti varijable **a**, odnosno **as**.

Prva ugniježđena **if** naredba (koja se nalazi unutar **then** dijela početne **if** naredbe) provjerava drugi predznak, odnosno vrijednost varijable **pr2**. Ako je **pr2=0**, to predstavlja predznak plus, te će on i biti ispisan (između varijabli **as** i **bs**), a ako nije onda će biti ispisan minus (između varijabli **as** i **bs**). Naredba **join** spaja nekoliko stringova u jedan string, te će oni biti ispisani putujući jedan za drugim po ekranu.



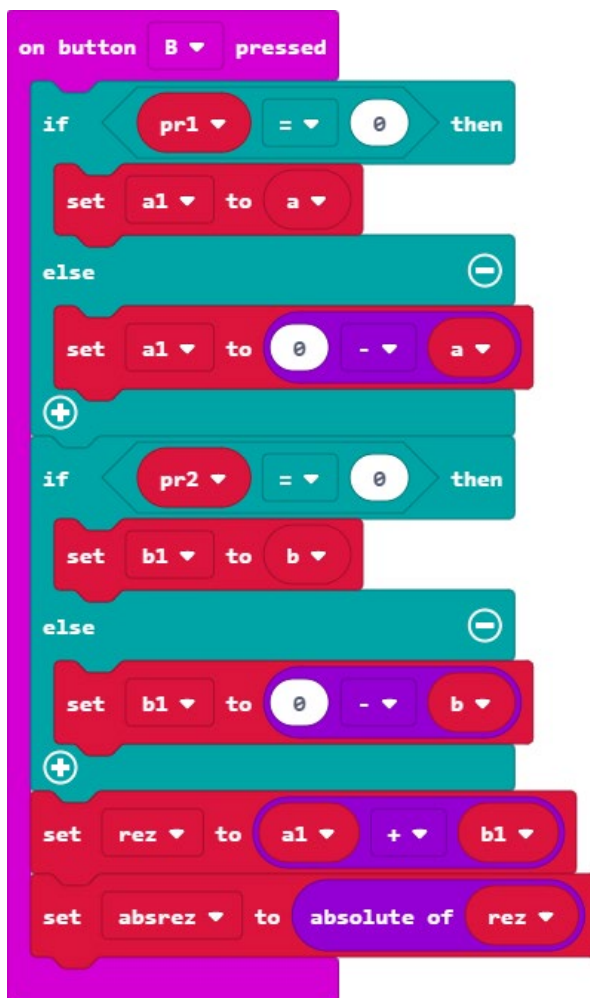
6. korak

U dijelu **else** (od početne/glavne **if** naredbe) nalaze se naredbe koje će se izvesti ako **pr1** ne bude jednak 0 (što znači da će on predstavljati predznak minus) i ako **a** nije 0. U tom slučaju će predznak minus trebati biti ispisan. Stoga naredbe za ispis unutar tog **else** dijela počinju s “-“. Ugniježđena **if then else** naredba unutar toga opet provjerava što je s **pr2**, te ovisno o njegovoj vrijednosti ispisuje plus ili minus (između varijabli **as** i **bs**). Prvo ćete unutar **on button B pressed** naredbe definirati izračunavanje varijable **rez**, koja će predstavljati rezultat računa koji se prikazuje pritiskom na tipku A.

Prvom **if then else** naredbom definirate varijablu **a1** koja postaje jednaka varijabli **a** u slučaju ako je **pr1=0** (tj. ako je prvi predznak pozitivan), odnosno jednaka **-a** ako **pr1** nije jednak nuli (tj. ako je prvi predznak negativan).

Slično definirate i varijablu **b1**, u ovisnosti o varijablama **pr2** i **b**. Konačno, varijablu **rez** definirate kao **a1+b1**, dakle **rez= a1+b1**.

Ujedno definirate i varijablu **absrez** kao apsolutnu vrijednost varijable **rez**.



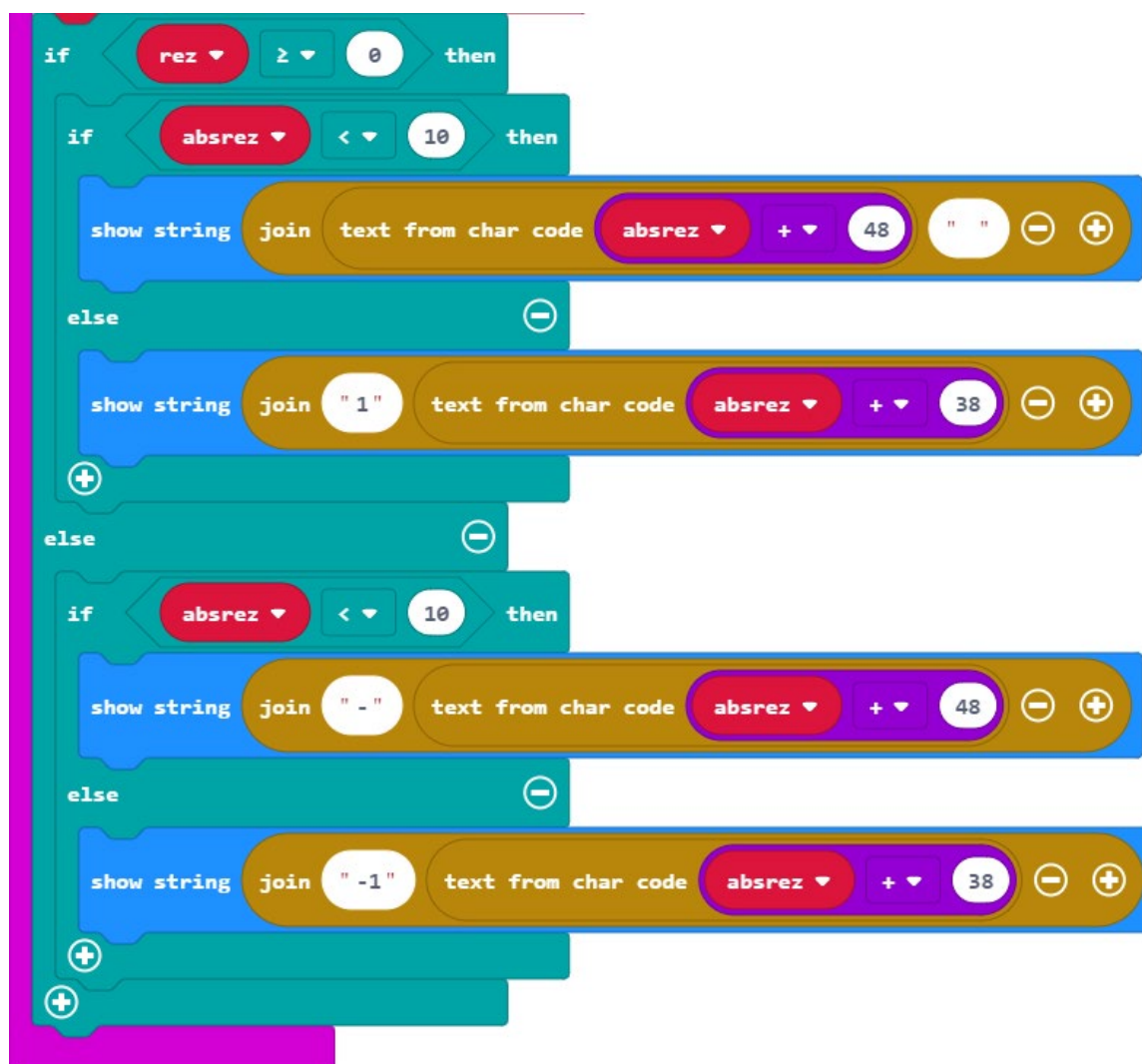
7. korak

Sad treba ispisati vrijednost varijable **rez**, ali tako da se ispiše kao string varijabla.

Ako je **rez** negativan, morat ćete ispisati prvo “-”, a zatim apsolutnu vrijednost. Ako je **rez** pozitivan, onda nema ispisivanja nikakvog predznaka. Za ispitivanje i pravovaljani ispis koristi ćete **if then else** naredbu.

No, ovdje imate još jednu “komplikaciju”. Naime, vidjeli ste gore da broj pretvarate u string tako da broju dodate 48 i na to primijenite naredbu **text from char code**. To vrijedi za jednoznamenkaste brojeve. Prije ste imali upravo takve, međutim ovdje rezultat može biti i dvoznamenkast, **rez** može poprimiti vrijednosti od -18 do 18. Stoga, u slučaju kad je **rez** dvoznamenkast (što ispitujete sa **if apsrez<10**), trebat će ispisati prvo znamenku “1”, a zatim od **apsrez** oduzeti 10 (da vam ostane druga znamenka), te njoj dodati 48 i primijeniti naredbu **text from char code**. Naravno, umjesto oduzimanja broja 10 i zatim dodavanja broja 48, to ćete skratiti na dodavanje broja 38.

Ugniježđeni niz upravo opisanih naredbi izgleda ovako:



Sve što se treba dogoditi pritiskom na tipku B prikazano je ovdje:

```
on button B pressed
  if pr1 = 0 then
    set a1 to a
  else
    set a1 to 0 - a
  if pr2 = 0 then
    set b1 to b
  else
    set b1 to 0 - b
  set rez to a1 + b1
  set absrez to absolute of rez
  if rez >= 0 then
    if absrez < 10 then
      show string join text from char code absrez + 48 " "
    else
      show string join "1" text from char code absrez + 38
    else
      if absrez < 10 then
        show string join "- " text from char code absrez + 48
      else
        show string join "-1" text from char code absrez + 38
```

Time ste iskodirali sve željeno.

Gotov program

The program consists of three main event-driven blocks triggered by button presses:

- on button B pressed:**
 - Initializes `pr1` to 0.
 - Asks for input `a` and sets `a1` to `a`.
 - Initializes `pr2` to 0.
 - Asks for input `b` and sets `b1` to `b`.
 - Calculates `rez = a1 + b1`.
 - Calculates `absrez = absolute of rez`.
 - Checks if `rez >= 0`. If true, displays `absrez` followed by a space and a quote.
 - Checks if `absrez < 10`. If true, displays `"1"` followed by `absrez` and a quote.
 - Otherwise, displays `"-1"` followed by `absrez` and a quote.
- on button A+B pressed:**
 - Initializes `pr1` to a random number between 0 and 2.
 - Asks for input `a` and sets `a` to that input.
 - Initializes `pr2` to a random number between 0 and 2.
 - Asks for input `b` and sets `b` to that input.
 - Displays the LED matrix.
 - Displays the LED matrix.
 - Displays the LED matrix.
 - Pauses for 100ms.
 - Clears the screen.
 - Calculates `as = text from char code a + 48`.
 - Calculates `bs = text from char code b + 48`.
- on button A pressed:**
 - Checks if `pr1 = 0` or `a = 0`. If true:
 - Checks if `pr2 = 0`. If true, displays `as "+" bs`.
 - Otherwise, displays `as "-" bs`.
 - Otherwise:
 - Checks if `pr2 = 0`. If true, displays `"-" as "+" bs`.
 - Otherwise, displays `"-" as "-" bs`.