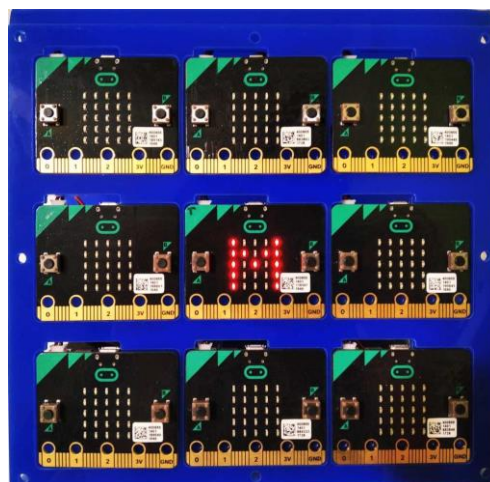


Memory – micro:bit

Izradite igru memory koristeći 9 micro:bitova. Potrebno je izraditi 2 programa – jedan za micro:bitove koji sudjeluju u igri (8 micro:bitova) i prikazuju sličice i jedan za glavni (master) micro:bit koji nasumično odredi sličice na ostalim micro:bitovima.

Igra se sastoji od 4 para sličica koje se tijekom igre pokušavaju pronaći aktiviranjem dva po dva micro:bitova na kojima se prikaže sličica pritiskom na tipkalo A. Micro:bitovi radio vezom provjeravaju jesu li sličice iste ili ne. U slučaju da sličice čine par, obje sličice moraju dva puta zatreperti i ostati prikazane dok se ne pokrene nova igra. Ako su sličice različite ili igrač nije unutar 2 sekunde aktivirao drugi micro:bit, neka se na ekranu prikaže znak X dva puta i onda ugasi.



Prije svake nove igre potrebno je pomoću master micro:bita resetirati ostale micro:bitove (tipkalo B) te nasumično raspodijeliti sličice na 8 micro:bitova (tipkalo A). Tada su ostali micro:bitovi spremni za igru.

Opis programa

Pritiskom na tipkalo B na master micro:bitu resetiraju se vrijednosti na ostalim micro:bitovima.

Pritiskom na tipkalo A na master micro:bitu nasumično se dodijele sličice ostalim micro:bitovima.

Pritiskom na tipkalo A na micro:bitu koji sudjeluje u igri prikazuje se sličica na ekranu tog micro:bita te micro:bit radio vezom šalje broj (specifičan za njegovu sličicu) drugim micro:bitovima.

Sličica ostaje na ekranu ako čini par sa sličicom nekog drugog micro:bita, inače se na ekranu prikaže X i ekran se obriše.

Napomena: u igri Memory se smiju aktivirati samo dva micro:bita istovremeno i tek kada su micro:bitovi iskomunicirali čine li sličice na njima par ili ne, igrač može aktivirati druga dva micro:bita.

Program se izrađuje u MakeCode editoru kojem možete pristupiti preko poveznice: <https://makecode.microbit.org/>.

Izrada programa – master micro:bit

1. korak

Pri pokretanju master micro:bita želite da se kreiraju i postave elementi koji će se koristiti u ostalim dijelovima programa, zato odaberite **on start** blok i stavite ga u prostor za programiranje. Kako bi master micro:bit mogli raspoznati od drugih micro:bitova, označite slovo M u naredbi **show leds**.

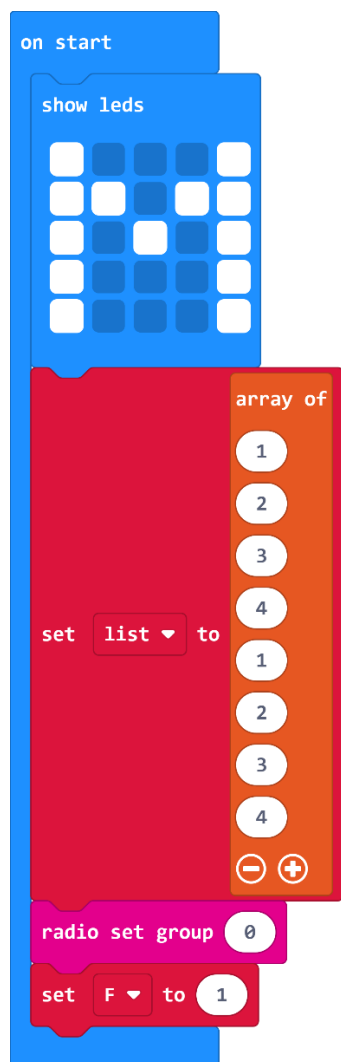
Master micro:bit komunicira s 8 micro:bitova i njima nasumično šalje sličice. Tih 8 micro:bitova bit će predstavljeni jednom listom naredbom **set list to** iz **Arrays** kategorije naredbi. Svaki element u listi predstavlja jedan micro:bit koji će se koristiti u igri. Kako će svaka dva micro:bita imati istu sličicu, elemente liste imenujte brojevima 1-4, a svaki broj će se duplo koristiti. Pritiskom na znak plus, dodajete elemente liste dok ih ne bude 8.

Postavite grupni ID (radio kanal) za radio komunikaciju naredbom **radio set group** iz kategorije naredbi **Radio**. I upišite broj 0 kao kanal na kojem će master micro:bit komunicirati sa svim ostalim micro:bitovima. Micro:bitovi

Memory – micro:bit

moгу međusobno primati i slati poruke preko radio veze samo ako se nalaze na istom kanalu, a jedan micro:bit može slušati samo jedan ID grupe u bilo kojem trenutku.

U kategoriji **Variables**, opcijom **Make a Variable**, kreirajte varijablu **F**. Vrijednost kreirane varijable **F** postavite na 1 naredbom **set F to 1**. Kasnije u programu, varijabla **F** će predstavljati druge radio kanale kada će master micro:bit trebati komunicirati samo s određenim micro:bitovima. Za ostale radio kanale će se koristiti brojevi 1-4 i zato je sada varijabla **F** jednaka 1.



2. korak

Pritiskom na tipkalo A, master micro:bit mora nasumično dodijeliti i poslati parove sličica ostalim micro: bitovima pa iz kategorije **Input** odaberite **on button A pressed** naredbu. Želite da se sličice na početku svake nove igre rasporede po različitim micro: bitovima. Kako biste svaki puta imali drukčiji raspored micro:bitova potrebno je nasumično sortirati listu iz prvog koraka. Iskoristite takozvani [Fisher-Yates algoritam](#) za sortiranje elemenata liste. Ideja algoritma:

- uzima listu elemenata koju treba nasumično sortirati
- nasumično odabere jedan element liste koji još nije sortiran
- zamijeni taj element sa zadnjim elementom u listi koji još nije bio odabran
- ponavlja 2. i 3. točku sve dok svi elementi nisu sortirani

Memory – micro:bit

Opcijom **Make a Variable** kreirajte tri varijable: **i**, **j** i **t**.

Iz kategorije **Variables**, naredbom **set i to** postavite vrijednost varijable **i** na duljinu liste umanjenu za 1 tako da kao argument naredbe stavite naredbu oduzimanja iz **Math** kategorije te u desno polje upišete broj 1, a u lijevo stavite **length of array list** iz **Arrays** kategorije naredbi. Duljina liste je broj elemenata liste, što je 8 u ovom slučaju. Svakom elementu liste možete pristupiti zasebno, preko indeksa: prvi element liste ima indeks 0, a zadnji element liste ima indeks duljina liste umanjena za 1. Dakle, indeksi liste su 0-7 te trenutno varijabla **i** ima vrijednost 7 i pomoću nje se krećete po listi, od zadnjeg indeksa liste prema prvom.

Ostatak algoritma se nalazi u **while do** petlji koju možete pronaći u **Loops** kategoriji. Naredbe unutar petlje se ponavljaju sve dok je varijabla **i** veća ili jednaka od nule. To jest, sve dok ne prođete svim elementima liste.

Naredbom **set j to** vrijednost varijable **j** postavite na ostatak pri dijeljenju (**remainder of**) nasumično odabranog broja između 0 i 7 (**pick random 0 to length of array list – 1**) i broja **i+1**. Ova varijabla može poprimiti bilo koji indeks liste od 0-7, ovisno o varijabli **i**. Naredbe uzmite iz **Math** kategorije.

Varijabla **t** služi kao privremena varijabla u koju privremeno pohranjujete element liste dok ne zamijenite elemente. Vrijednost varijable **t** postavite na element liste pod indeksom **i** tako da kao argument naredbe **set t to** stavite **list get value at i**.

Za zamjenu dva elementa liste iskoristite dva puta naredbu **list set value at** iz **Arrays** kategorije. Element liste pod indeksom **i** postaje element liste pod indeksom **j** i element liste pod indeksom **j** postaje varijabla **t** u koju ste već ranije pohranili element liste pod indeksom **i**.

Na kraju umanžite vrijednost varijable **i** za 1 naredbom **set i to i-1**. Ovom naredbom ćete redom ići od zadnjeg indeksa pa do prvog indeksa liste.

```
on button A pressed
  set i to length of array list - 1
  while i >= 0
    do
      set j to remainder of pick random 0 to length of array list - 1 ÷ i + 1
      set t to list get value at i
      list set value at i to list get value at j
      list set value at j to t
      set i to i - 1
```

3. korak

Sada ste sortirali listu i tako sortiranoj listi želite dodijeliti sličice. Za 8 micro:bitova su potrebne 4 vrste sličica, svaka sličica se šalje dva puta. Želite da se jedna vrsta sličice šalje na micro:bitove koji su u listi označeni brojem 1, druga vrsta sličice onima u listi označenima brojem 2, itd. Na ovaj način ste osigurali da će jedna sličica biti poslana na dva micro:bitova.

Memory – micro:bit

Petlja **for element value of list** iz kategorije **Loops** će izvršiti niz naredbi za svaki element u listi. Lista koju koristi je ona koju ste maloprije sortirali. Dakle, naredbe će se izvršavati za prvi element novo sortirane liste do zadnjeg elementa liste.

Postavite radio kanal na varijablu **F (radio set group F)**. Varijabla F je u **on start** bloku postavljena na 1. Master micro:bit će sa svakim micro:bitom imati zasebnu komunikaciju na vlastitom radio kanalu od 1-8 kada će svakom poslati poruku.

4 **if** petlje iz kategorije **Logic** dodjeljuju 4 različite sličice ovisno o elementu liste 1-4. Svakom elementu liste poslat će se jedna poruka radio vezom naredbom **radio send message1** iz **Radio** kategorije naredbi. Klikom na strelicu pored naziva **message1** otvara se padajući izbornik u kojem odaberite opciju **Add a new msg**. Otvorit će se prozor **New msg**: u koji upišite naziv nove poruke. Nove poruke imenujte KVADRAT, SRCE, TROKUT i CVIJET. Svaka poruka ima naziv sličice koju će taj micro:bit prikazivati: za vrijednost 1, 2, 3 ili 4 šalje se poruka KVADRAT, SRCE, TROKUT ili CVIJET.

Naredbom **change F by 1** iz kategorije **Variables** mijenjate radio kanal za 1. Nakon izlaska iz petlje **for element value of list** i nakon što su svim micro: bitovima poslana poruke, stavite radio kanal na 0 naredbom **radio set group 0**. Sada master micro:bit komunicira samo sa micro: bitovima na tom kanalu, kojeg koristite kada želite da komunicira sa svima istovremeno.

Memory – micro:bit

```
on button A pressed
  set i to length of array list - 1
  while i >= 0
    do
      set j to remainder of pick random 0 to length of array list - 1 ÷ i + 1
      set t to list get value at i
      list set value at i to list get value at j
      list set value at j to t
      set i to i - 1
  end while
  for element value of list
    do
      radio set group F
      if value = 1 then
        radio send KVADRAT
      end if
      if value = 2 then
        radio send SRCE
      end if
      if value = 3 then
        radio send TROKUT
      end if
      if value = 4 then
        radio send CVIJET
      end if
      change F by 1
    end do
  end for
  radio set group 0
```

4. korak

Na kraju još trebate isprogramirati da se pritiskom na tipkalo B resetiraju sve vrijednosti na ostalih 8 micro:bitova kako bi bili spremni za ponovnu nasumičnu raspodjelu sličica pritiskom na tipkalo A. U **on button B pressed** blok stavite naredbu **radio send message1** iz kategorije **Radio** te dodajte poruku RESTART opcijom **Add a new msg.** Kada ostali micro: bitovi prime tu poruku, izvršit će se niz naredbi koje su zadane u njihovom programu. Nakon toga postavite vrijednost varijable **F** na početnu vrijednost 1.

Memory – micro:bit

```
on button B pressed
  radio send RESTART
  set F to 1
```

Gotov program za master micro:bit

```
on start
  show leds
  array of
    1
    2
    3
    4
  set list to
    1
    2
    3
    4
  radio set group 0
  set F to 1

on button B pressed
  radio send RESTART
  set F to 1

on button A pressed
  set i to length of array list - 1
  while i >= 0
    do
      set j to remainder of pick random 0 to length of array list - 1 ÷ i + 1
      set t to list get value at i
      list set value at i to list get value at j
      list set value at j to t
      set i to i - 1
  for element value of list
    do
      radio set group F
      if value = 1 then
        radio send KVADRAT
      if value = 2 then
        radio send SRCE
      if value = 3 then
        radio send TROKUT
      if value = 4 then
        radio send CVIJET
      change F by 1
  radio set group 0
```

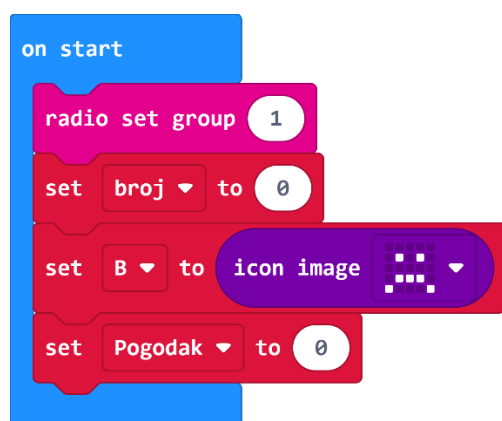
Izrada programa – ostali micro:bitovi

Memory – micro:bit

Napomena: Svaki micro:bit koji sudjeluje u igri ima jedinstvenu radio grupu. Na dva mjesta u programu (1. korak i 2. korak) potrebno je, prije prebacivanja programa na micro:bit, postaviti radio grupu na jedinstven broj 1-8 za svaki micro:bit.

1. korak

Pri pokretanju micro:bita, u **on start** bloku, želite da se kreiraju i postave elementi koji će se koristiti u ostalim dijelovima programa. Postavite radio kanal na 1. Ovaj radio kanal je jedinstven za svaki micro:bit uključen u igru i omogućava jedan na jedan komunikaciju s master micro:bitom. Opcijom **Make a Variable** u kategoriji **Variables** kreirajte varijablu **broj** i postavite je na 0. Ova varijabla omogućava micro: bitovima da iskomuniciraju jesu li otkrivene sličice tijekom igre jednake ili različite. Vrijednost varijable **broj** može biti broj 1, 2, 3 ili 4. Kreirajte još varijable **B** i **Pogodak**. Varijablu **B** postavite na sličicu tužnog emotikona koja se prikazuje ako micro:bitu nije dodijeljena sličica za igru, a varijablu **Pogodak** postavite na 0. Ovom varijablom će se u programu onemogućiti da aktivirani micro: bitovi koji već čine par, slučajnim pritiskom tipkala A, pošalju radio vezom broj.



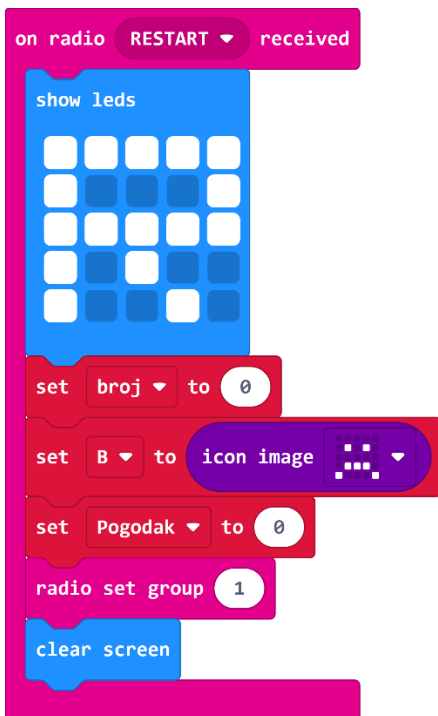
2. korak

Kada micro:bit primi poruku RESTART od master micro:bita, sve varijable vraća na početno stanje kako bi bio spreman za ponovnu nasumičnu raspodjelu sličica. Naredbom **show leds** postavite da se na ekranu kratko ispiše slovo R što označava da je micro:bit u fazi resetiranja. Kada na ekranu više nema slova R, micro:bit je resetiran. Naredbu stavite u blok **on radio message1 received** iz kategorije Radio. Klikom na strelicu pored naziva **message1** otvara se padajući izbornik u kojem odaberite opciju **Add a new msg**. Otvorit će se prozor **New msg**: u koji upišite naziv nove poruke RESTART.

Iz kategorije **Variables**, naredbom **set to** postavite varijablu **broj** na 0. Ova varijabla omogućava micro: bitovima da iskomuniciraju jesu li otkrivene sličice tijekom igre jednake ili različite. Varijabla **B** neka bude sličica tužnog lica koja se prikazuje ako micro:bitu nije dodijeljena sličica za igru. Varijablu **Pogodak** postavite na 0.

Radio kanal postavite na broj 1 naredbom **radio set group**. Ovaj radio kanal je jedinstven za svaki micro:bit uključen u igru i omogućava jedan na jedan komunikaciju s master micro:bitom. U igri se koristi osam micro:bitova pa ćete imati radio kanale u rasponu od 1 do 8.

Memory – micro:bit

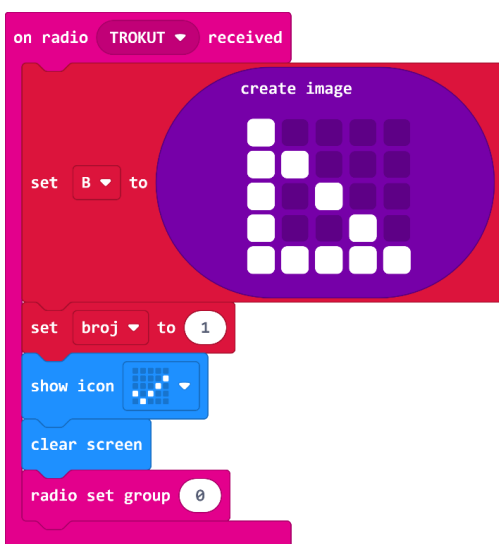


3. korak

Ovisno koju poruku za sličicu micro:bit primi (TROKUT, KVADRAT, CVIJET ili SRCE), izvršava se niz radnji specifičan za poruku koju je dobio. Za svaku poruku potrebno je napisati zaseban program.

Prvo izradite program za poruku TROKUT. Iz kategorije **Radio** odaberite naredbu **on radio message1 received**. Opcijom **Add a new msg** dodajte poruku TROKUT. Sada imate blok **on radio TROKUT received** i u njega stavite nekoliko naredbi koje će se izvršiti kada taj micro:bit primi poruku TROKUT od master micro:bita.

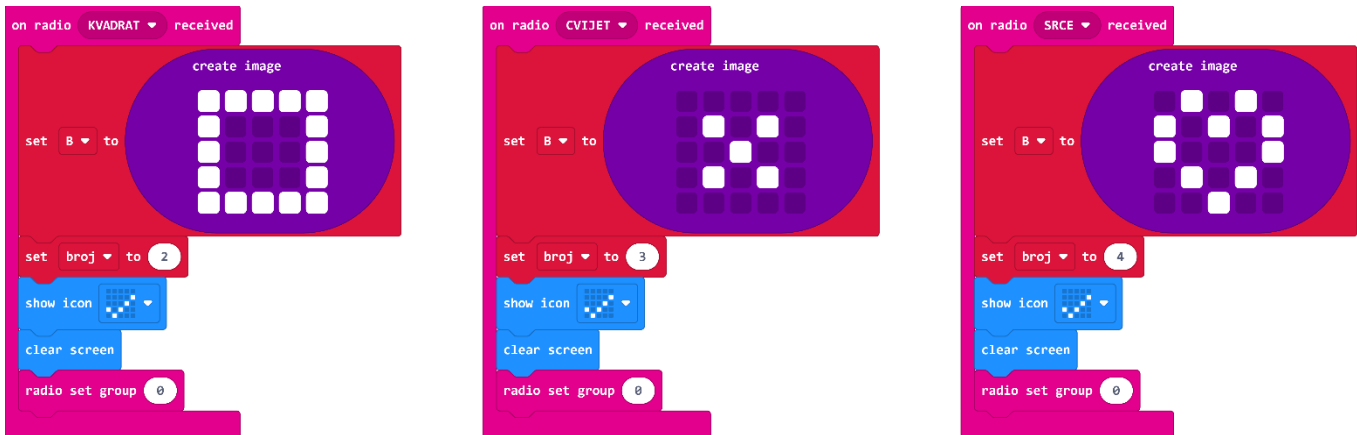
Naredbom **set B to** postavite varijablu **B** na sličicu koja opisuje poruku koju je micro:bit dobio. Sličicu izradite pomoću naredbe **create image** iz kategorije **Images**. U ovom slučaju je to trokut. Varijablu **broj** postavite na 1. Na ekranu neka se pokaže kvačica kada je micro:bitu uspješno dodijeljena sličica naredbom **show icon**, a potom neka se ekran obriše naredbom **clear screen**. Naredbom **radio set group** stavite radio kanal na 0. Na ovom radio kanalu micro:bit može komunicirati sa svim ostalim micro: bitovima.



Memory – micro:bit

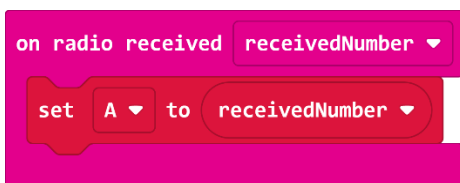
4. korak

Za svaku primljenu poruku potrebno je napisati zaseban program. Duplicirajte blok **on radio TROKUT received** i izmijenite naziv poruke, sličicu i varijablu **broj** kako biste složili programe koji se izvršavaju primanjem poruka KVADRAT, CVIJET i SRCE. Postoje samo dva micro:bitova koja će imati istu sličicu, a time i istu varijablu **broj**. Oni čine jedan par. Za svaku sličicu morate staviti različiti broj.



5. korak

Kako bi micro: bitovi mogli usporediti sličice, međusobno si pošalju broj radio signalom koji je specifičan za svaku vrstu sličice. Primljeni broj neka spremaju u novu varijablu **A**. Nakon kreiranja varijable A, postavite je na **receivedNumber** koji povucite direktno iz bloka **on radio received receivedNumber**. Pod varijablu **A** se sprema broj koji je poslan s drugog micro:bita, a koji predstavlja vrstu sličice na tom micro:bitu.



6. korak

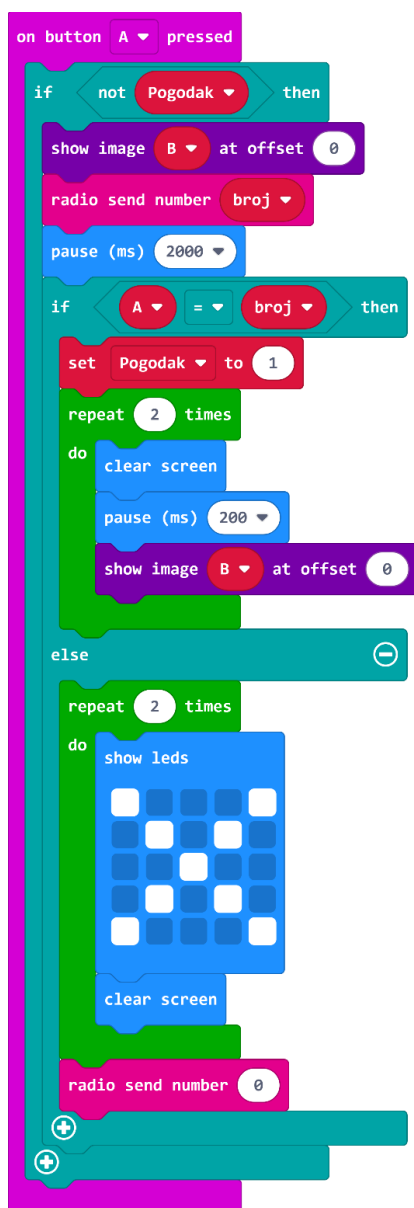
Pritiskom na tipkalo A na micro:bitu prikazat će se sličica na ekranu i poslat će se broj specifičan za sličicu koju prikazuje. Tijekom igre se uvijek aktiviraju samo dva micro:bitova i oni trebaju iskomunicirati brojeve koji predstavljaju vrstu sličice. Usporedbom brojeva provjerava se čine li micro:bitovi jedan par ili ne.

U blok **on button A pressed** stavite naredbu **show image B at offset 0** iz kategorije naredbi **Images** kako bi se prikazala sličica koja je micro:bitu dodijeljena i spremljena pod varijablom **B**. Nakon toga neka micro:bit pošalje svoj broj (specifičan za njegovu sličicu) drugim micro:bitovima naredbom **radio send number broj**. Prije nego micro:bit krene uspoređivati sličice, naredbom **pause** stavite vremenski period od 2 sekunde da razmijeni radio signal s drugim micro:bitom.

Sada slijedi usporedba varijabli **A** i **broj**. Iz kategorije **Logic** uzmite **if then else** naredbu i kao uvjet koji se provjerava stavite naredbu jednakosti i varijable **A** i **broj**. Ako je primljeni broj koji predstavlja sličicu drugog micro:bita

Memory – micro:bit

(spremljen pod varijablom **A**) jednak varijabli **broj** koja predstavlja njegovu sličicu, u **if** dijelu naredbama **repeat**, **clear screen**, **pause** i **show image B at offset 0** postavite da sličica zatreperi dva puta i ostane na ekranu. Tada dva odabrana micro:bitova čine par. Inače, u **else** dijelu, neka se naredbama **repeat**, **show leds** i **clear screen** na ekranu prikaže znak X dva puta i neka se ekran očisti. Tada dva odabrana micro:bitova ne čine par.



Gotov program za ostale micro:bitove

Memory – micro:bit

```
on start
  radio set group 1
  set broj to 0
  set B to icon image
  set Pogodak to 0
```

```
on radio received receivedNumber
  set A to receivedNumber
```

```
on radio RESTART received
  show leds
  set broj to 0
  set B to icon image
  set Pogodak to 0
  radio set group 1
  clear screen
```

```
on radio KVADRAT received
  create image
  set B to
  set broj to 2
  show icon
  clear screen
  radio set group 0
```

```
on button A pressed
  if not Pogodak then
    show image B at offset 0
    radio send number broj
    pause (ms) 2000
  if A == broj then
    set Pogodak to 1
    repeat 2 times
      do
        clear screen
        pause (ms) 200
        show image B at offset 0
  else
    repeat 2 times
      do
        show leds
        clear screen
    radio send number 0
```

```
on radio CVIJET received
  create image
  set B to
  set broj to 3
  show icon
  clear screen
  radio set group 0
```

```
on radio TROKUT received
  create image
  set B to
  set broj to 1
  show icon
  clear screen
  radio set group 0
```

```
on radio SRCE received
  create image
  set B to
  set broj to 4
  show icon
  clear screen
  radio set group 0
```