



IRIM - Institut za razvoj  
i inovativnost mladih

micro:bit u nastavi

# Vježbe za č/ć i ije/je

RAZREDNA NASTAVA



HRVATSKI JEZIK

## Vježbe za č/ć i ije/je

Izradite program pomoću kojeg se na micro:bitu može vježbati č/ć ili ije/je.

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

<https://bit.ly/vjezbe-c-ije>.

## Korištenje s učenicima

<b>Predmet</b>	Hrvatski jezik
<b>Razred</b>	2. i 3., osnovna škola
<b>Polugodište</b>	1. i 2.
<b>Odgojno-obrazovni ishodi</b>	<p><b>A. 2. 4</b> Učenik piše školskim rukopisnim pismom slova, riječi i kratke rečenice u skladu s jezičnim razvojem.</p> <p><b>A. 3. 4</b> Učenik piše vođenim pisanjem jednostavne tekstove u skladu s temom.</p>
<b>Ideja i izrada materijala</b>	Julija Vejić

## Opis programa

Duljim pritiskom na tipku A bira se između vježbe za č/ć i vježbe za ije/je.

Duljim pritiskom na tipku B potvrđuje se izbor vježbe.

**Vježba č/ć:** na micro:bitu se ispisuju riječi u kojima je pritiskom na tipku A ili B potrebno odrediti pišu li se sa ć ili č. Tipka A = ć, tipka B = č.

**Vježba ije/je:** na micro:bitu se ispisuju riječi u kojima je pritiskom na tipku A ili B potrebno odrediti je li riječ napisana ispravno ili ne. Tipka A = ispravno, tipka B = neispravno.

U svakoj vježbi ispisat će se 10 riječi. Ovisno o brzini i točnosti odgovaranja, na kraju vježbe, ispisat će se zbroj osvojenih bodova

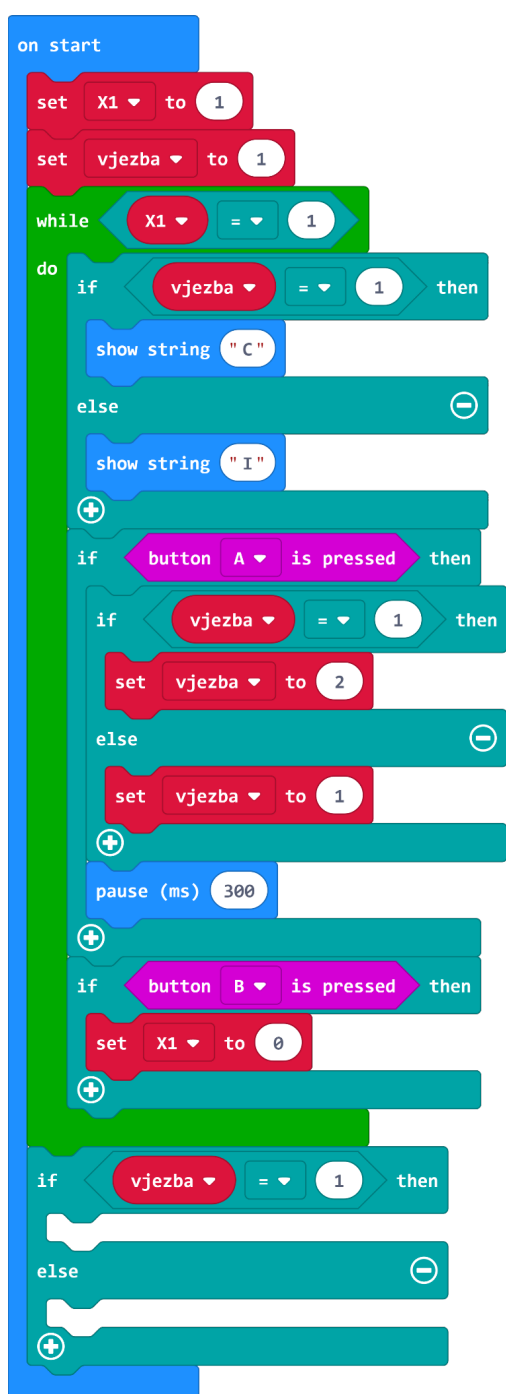
Program se izrađuje u MakeCode editoru kojem možete pristupiti preko poveznice:

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

## Izrada programa

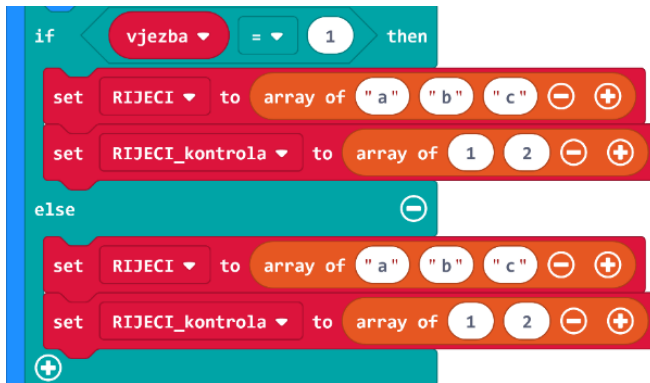
### 1. korak

Na početku programu u kategoriji **Variables – Make a variable**, potrebno je izraditi pomoćnu varijablu **X1** i varijablu **vjezba** koja će označavati je li odabrana vježba č/ć (**vjezba=1**) ili vježba ije/je (**vjezba=2**) na micro:bitu. Unutar bloka **on start** obje varijable postavite na vrijednost 1 te dodajte petlju **while do** iz kategorije **Loops** i postavite uvjet **X1=1**. Unutar te petlje pomoću naredbi **if then else** i **if else** iz kategorije **Logic** kreirajte niz naredbi koje će omogućiti da se pritiskom na tipku A bira između vježbi 1 i 2, a pritiskom na tipku B se potvrdi željena vježba. Potvrdom vježbe, varijabla **X1** se postavi na 0 čime se izađe iz petlje **while do**. Nakon petlje **while do**, stavite naredbu **if then else**.



## 2. korak

Nakon odabira vježbe, potrebno je kreirati listu riječi koje će se koristiti unutar svake vježbe. Osim liste riječi, kreirat ćete i listu brojeva za kontrolu ispravnosti tih riječi. Iz kategorije **Arrays** odaberite naredbu **set text list to** te ime varijable **text list** promijenite u **RIJECI**. Iz iste kategorije odaberite naredbu **set list to** te ime varijable **list** promijenite u **RIJECI\_kontrola**. Objе liste stavit ćete i u **then** dio i u **else** dio naredbe **if then else**, ali s različitim sadržajem riječi ovisno o vježbi. Pritiskom na plus možete dodavati elemente liste.



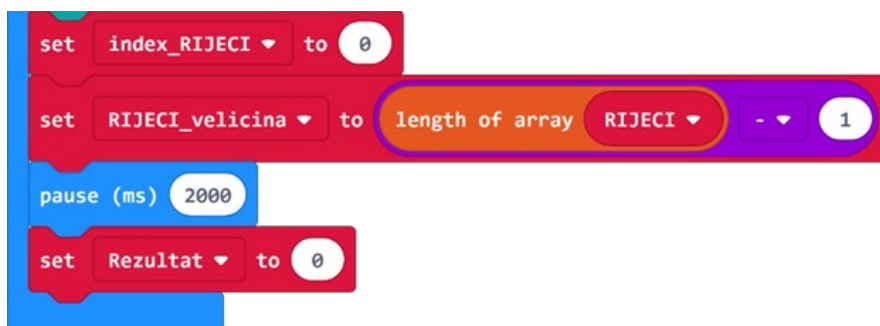
**Then** dio naredbe **if then else** se izvršava ako je varijabla **vjezba** jednaka 1 pa sadržaj liste **RIJECI** čine riječi za vježbu č/ć. Kao elemente liste upišite riječi koje želite koristiti u vježbi za č/ć te paralelno u listu **RIJECI\_kontrola**, na istu poziciju na kojoj je riječ koju ste upisali, upišite broj 1 ako ta riječ sadrži slovo ć ili broj 2 ako sadrži slovo č. Na primjer, za riječ BRATIC u listi **RIJECI** upisan je broj 1 u listi **RIJECI\_kontrola**. Za riječ IGRAC u listi **RIJECI** upisan je broj 2 u listi **RIJECI\_kontrola**. U ovom primjeru upisano je 78 riječi za vježbu č/ć pa su obje liste u **then** dijelu naredbe **if then else** duljine 78.



Ako vrijednost varijable **vjezba** nije jednaka 2 izvršava se **else** dio naredbe **if then else** pa sadržaj liste **RIJECI** čine riječi za vježbu ije/je. Kao elemente liste upišite riječi koje želite koristiti u vježbi za ije/je te paralelno u listu **RIJECI\_kontrola**, na istu poziciju na kojoj je riječ koju ste upisali, upišite broj 1 ako je ta riječ ispravno napisana ili broj 2 ako nije ispravno napisana. Na primjer, za riječ BIJEG u listi **RIJECI** upisan je broj 1 u listi **RIJECI\_kontrola**. Za riječ BJELAC u listi **RIJECI** upisan je broj 2 u listi **RIJECI\_kontrola**. U ovom primjeru upisano je 84 riječi za vježbu ije/je pa su obje liste u **else** dijelu naredbe **if then else** duljine 84.



Nakon naredbe **if then else**, kreirajte još tri varijable u kategoriji **Variables – Make a variable**. Varijabla **index\_RIJECI** predstavljat će poziciju neke riječi iz liste, nju postavite na 0. Varijabla **RIJECI\_velicina** predstavljat će poziciju zadnjeg elementa liste, nju postavite na duljinu liste **RIJECI** umanjene za 1 koristeći naredbu **length of array** iz kategorije **Arrays** (u ovom primjeru, pozicija zadnjeg elementa liste **RIJECI** je 77 za vježbu č/ć ili 83 za vježbu ije/je). Varijablu **Rezultat** postavite na 0.



### 3. korak

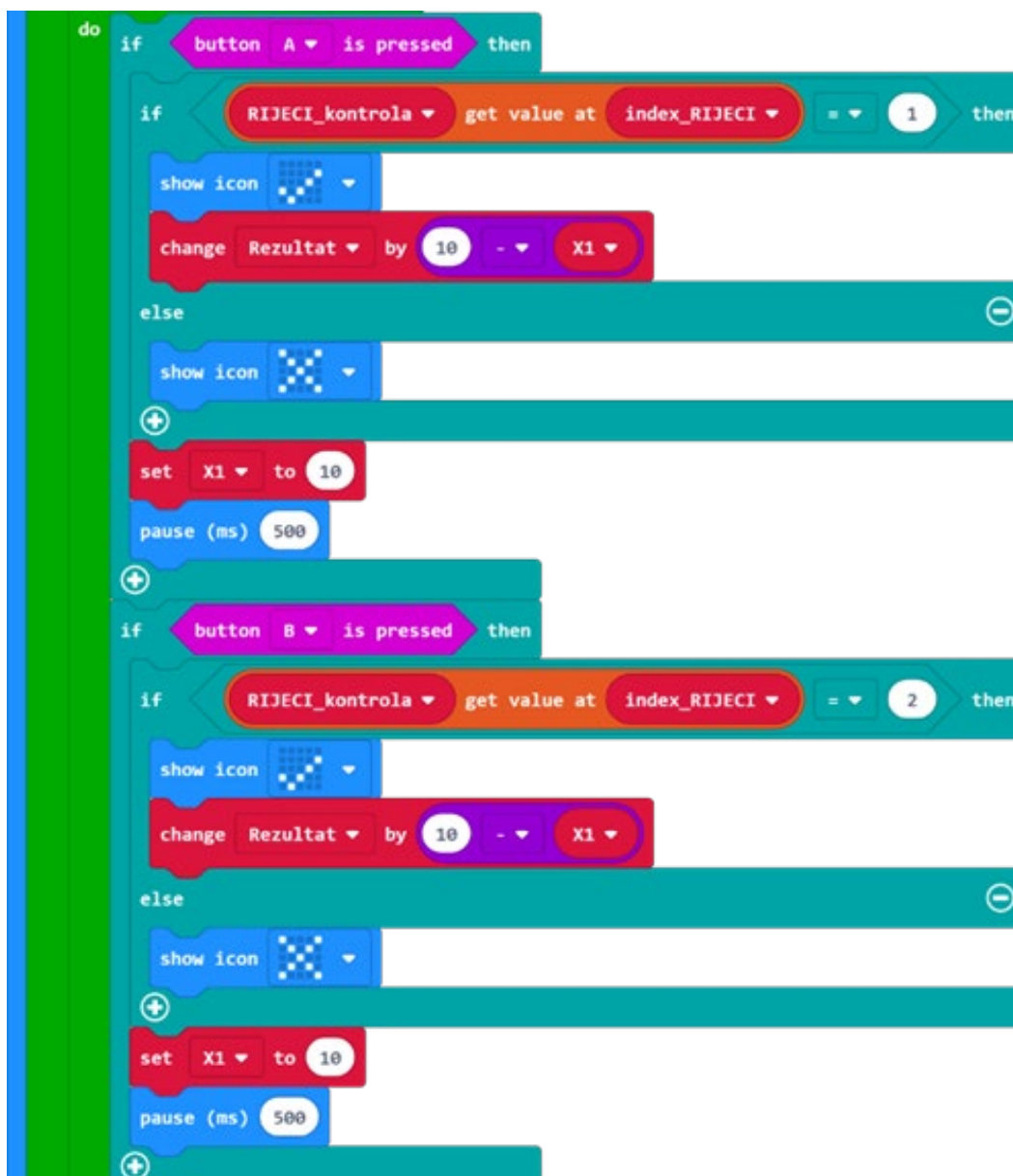
U **forever** blok stavite naredbu **for index 0 to 4** iz kategorije **Loops** te umjesto broja 4 upišite broj 9. Kreirajte varijablu **Pokusaj** i stavite je umjesto **index** u naredbi **for index 0 to 9**. Sve naredbe unutar te **for** petlje izvršavat će se 10 puta kao broj pokušaja za svaku vježbu. Nakon **for** petlje stavite naredbu **while true do** iz kategorije **Loops**, kojom se ponavlja naredba za prikaz konačnog rezultata spremljenog u varijabli **Rezultat**. Sada se vratite u **for** petlju i dodajte naredbe kojima će se na ekranu micro:bita, tijekom svakog pokušaja, prikazivati preostali broj pokušaja i nasumično odabrana riječ iz liste **RIJECI**. Za prikaz preostalog broja pokušaja koristite naredbu oduzimanja iz kategorije **Math** te oduzmite vrijednost varijable **Pokusaj** od broja 10. Za prikaz nasumično odabrane riječi koristite naredbu **get value at** iz kategorije **Arrays** te naredbu **pick random** iz kategorije **Math**. Sada dodajte naredbu **while do** iz kategorije **Loops** s uvjetom  $X1 < 10$  unutar koje će se izvršavati provjera odabranog odgovora korisnika tijekom svakog pokušaja u vježbi. To će se provjeravati pomoću dvije **if then** petlje, a s trećom **if then else** petljom će se ograničiti vrijeme davanja odgovora korisnika nakon čega se prelazi na idući pokušaj.

```
forever
  for Pokusaj from 0 to 9
  do
    set X1 to 0
    set index_RIJECI to pick random 0 to RIJECI_velicina
    clear screen
    show number 10 - Pokusaj
    pause (ms) 800
    clear screen
    show string RIJECI get value at index_RIJECI
    while X1 < 10
    do
      if button A is pressed then
        +
      if button B is pressed then
        +
      if X1 < 5 then
      else
        -
      +
      pause (ms) 300
      change X1 by 1
    while true
    do
      show number Rezultat
      pause (ms) 1000
      clear screen
```

The code is a Scratch script for a word guessing game on a Micro:bit. It is enclosed in a 'forever' loop. The main loop consists of a 'for' loop labeled 'Pokusaj' (Attempts) from 0 to 9. Inside this loop, the variable 'X1' is set to 0, and 'index\_RIJECI' is set to a random value between 0 and 'RIJECI\_velicina'. The screen is cleared, and the number of attempts (10 minus the current attempt) is shown. A 800ms pause follows, then the screen is cleared again, and the current letter (the value of 'RIJECI' at the 'index\_RIJECI' position) is shown. A 'while' loop runs as long as 'X1' is less than 10. Inside this 'while' loop, there are three 'if' statements: the first two check if button A or button B is pressed, each followed by an increment (+) block; the third checks if 'X1' is less than 5, followed by an 'else' block with a decrement (-) block. After these checks, there is another increment (+) block, a 300ms pause, and a 'change X1 by 1' block. After the 'while' loop, there is another 'while true' loop that shows the 'Rezultat' (Result), pauses for 1000ms, and clears the screen.

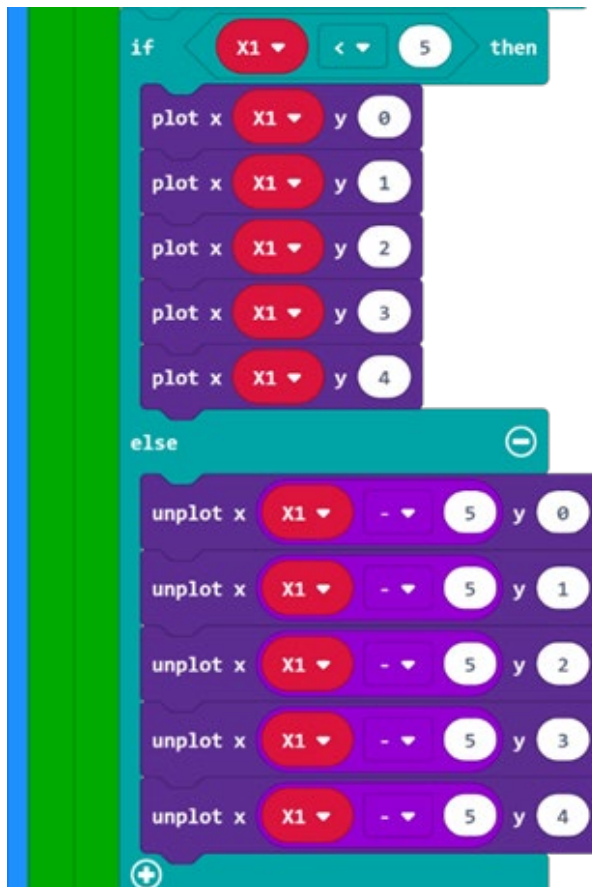
#### 4. korak

Naredba **if then** s uvjetom **button A is pressed** usporedit će vrijednost iz liste **RIJECI\_kontrola** na poziciji **index\_RIJECI** s brojem 1. Naredba **if then** s uvjetom **button B is pressed** usporedit će istu vrijednost, ali s brojem 2. U slučaju da je uvjet ispunjen, na ekranu se prikaže sličica ispravnog odgovora te se naredbom **change Rezultat by** iz kategorije **Variables**, ažurira rezultat ovisno o brzini odgovaranja koje je predstavljeno varijablom **X1**. Ako uvjet nije ispunjen, na ekranu se prikaže sličica netočnog odgovora.



#### 5. korak

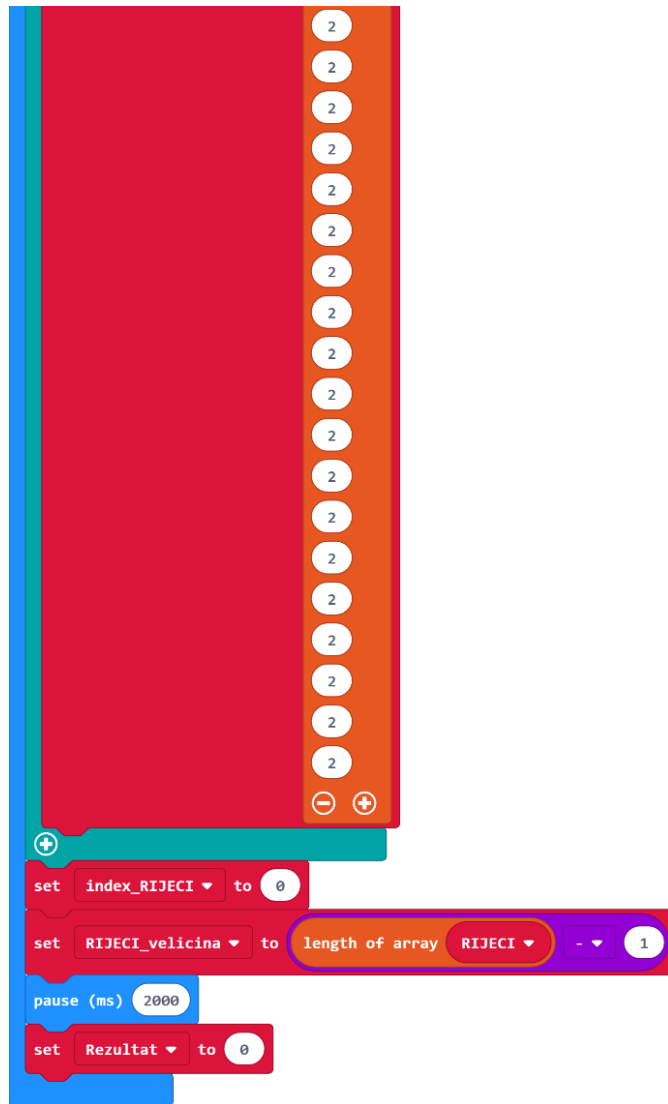
Protok vremena između svakog pokušaja zadan je varijablom **X1** čija je početna vrijednost 0. Vrijednost varijable **X1** će se povećavati sve dok se ne pritisne tipka A ili B. Kako vrijednost varijable **X1** raste, na ekranu će se paliti pa gasiti led lampice po stupcima, s lijeva na desno. Paljenje i gašenje lampica odvija se s lijeva na desno pomoću naredbi **plot x,y** i **unplot x,y** iz kategorije **Led**.











```
forever
  for Pokusaj from 0 to 9
  do
    set X1 to 0
    set index_RIJECI to pick random 0 to RIJECI_velicina
    clear screen
    show number 10 - Pokusaj
    pause (ms) 800
    clear screen
    show string RIJECI get value at index_RIJECI
    while X1 < 10
    do
      if button A is pressed then
        if RIJECI_kontrola get value at index_RIJECI = 1 then
          show icon [grid icon]
          change Rezultat by 10 - X1
        else
          show icon [grid icon]
          set X1 to 10
          pause (ms) 500
        if button B is pressed then
          if RIJECI_kontrola get value at index_RIJECI = 2 then
            show icon [grid icon]
            change Rezultat by 10 - X1
          else
            show icon [grid icon]
            set X1 to 10
            pause (ms) 500
```

