**ELEKTRO IN RAČUNALNIŠKA ŠOLA**

**Program:** [Tehnik / tehtnica računalništva](http://pters.scv.si/index.php?option=com_content&task=view&id=148&Itemid=141#_blank)

**Poklic: Tehnik / tehtnica računalništva**

###### POROČILO IZVEDENE VAJE PRI PREDMETU:

###### PRAKTIČNI POUK

Naslov vaje: Arduino Projekt

**Datum:** 28. 02. 2017

**Ime in priimek dijaka:** Domen Stropnik

**Razred in skupina:** 3.TRB – 2.Skupina

**Učitelj:** Klemen Hleb

**Ocena in podpis učitelja :**

##### Izdelano poročilo vsebuje naslednje točke:

1. Besedilo naloge Stran: \_\_\_\_\_\_\_
2. Shema in opis delovanja \_\_\_\_\_\_\_
3. Seznam elementov in komponent \_\_\_\_\_\_\_
4. Opis delovnih postopkov\_\_\_\_\_\_\_
5. Opis posameznih elementov in komponent \_\_\_\_\_\_\_
6. Komentar in analiza \_\_\_\_\_\_\_
7. Literatura in spletni naslovi \_\_\_\_\_\_\_
8. Varstvo pri delu \_\_\_\_\_\_\_
9. Pomisli, preizkusi in odgovori \_\_\_\_\_\_\_

**Kaj je Simon Says?**

***Simon Says*** je [elektronska spominksa](https://translate.googleusercontent.com/translate_c?depth=1&hl=sl&prev=search&rurl=translate.google.si&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Electronic_game&usg=ALkJrhi45ez2WufJHeIoGdZ1CAjuN_H1_g) igra, ki jo je izumil [***Ralph H. Bae*r**](https://translate.googleusercontent.com/translate_c?depth=1&hl=sl&prev=search&rurl=translate.google.si&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Ralph_H._Baer&usg=ALkJrhjuLAlo6XpJiXPZMzxBYknGKXvQSQ) in [***Howard J. Morrison***](https://translate.googleusercontent.com/translate_c?depth=1&hl=sl&prev=search&rurl=translate.google.si&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Howard_J._Morrison&usg=ALkJrhiTAEVQUIBJynQTUv_7Ka1-OEkZ_w), s pomočjo od [***Lenny Cope***](https://translate.googleusercontent.com/translate_c?depth=1&hl=sl&prev=search&rurl=translate.google.si&sl=en&sp=nmt4&u=https://en.wikipedia.org/w/index.php%3Ftitle%3DLenny_Cope%26action%3Dedit%26redlink%3D1&usg=ALkJrhgiXU--4jAbcyKUQPP6ntWpsJno3w)ki je sprogramiral igro. Naprava ustvari zaporedje tonov in luči, zahteva da uporabnik ponovi serijo z pritiskom na gumbe, po svojem spominu. Če uporabniku uspe, se serija nadeljuje in nastane daljša in bolj zapletena. Ko uporabniku ne uspe in zgrši, je konec igre. Simon Says je bil izdan leta 1978 v [Studiu 54](https://translate.googleusercontent.com/translate_c?depth=1&hl=sl&prev=search&rurl=translate.google.si&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Studio_54&usg=ALkJrhiDUErm7Cw0Un4TAmbOxV0rTInhvA), v [New Yorku](https://translate.googleusercontent.com/translate_c?depth=1&hl=sl&prev=search&rurl=translate.google.si&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/New_York_City&usg=ALkJrhicj5-YskzUrP9Rmv8vfspIqAuaxw) in je bil uspeh, saj je igrca postala pop kulturni simbol leta 1970 in 1980. {1}

[](https://en.wikipedia.org/wiki/File:Simon_game.jpg)

Slika 1: Prikaz igrce Simon

**Kje se uporablja?**

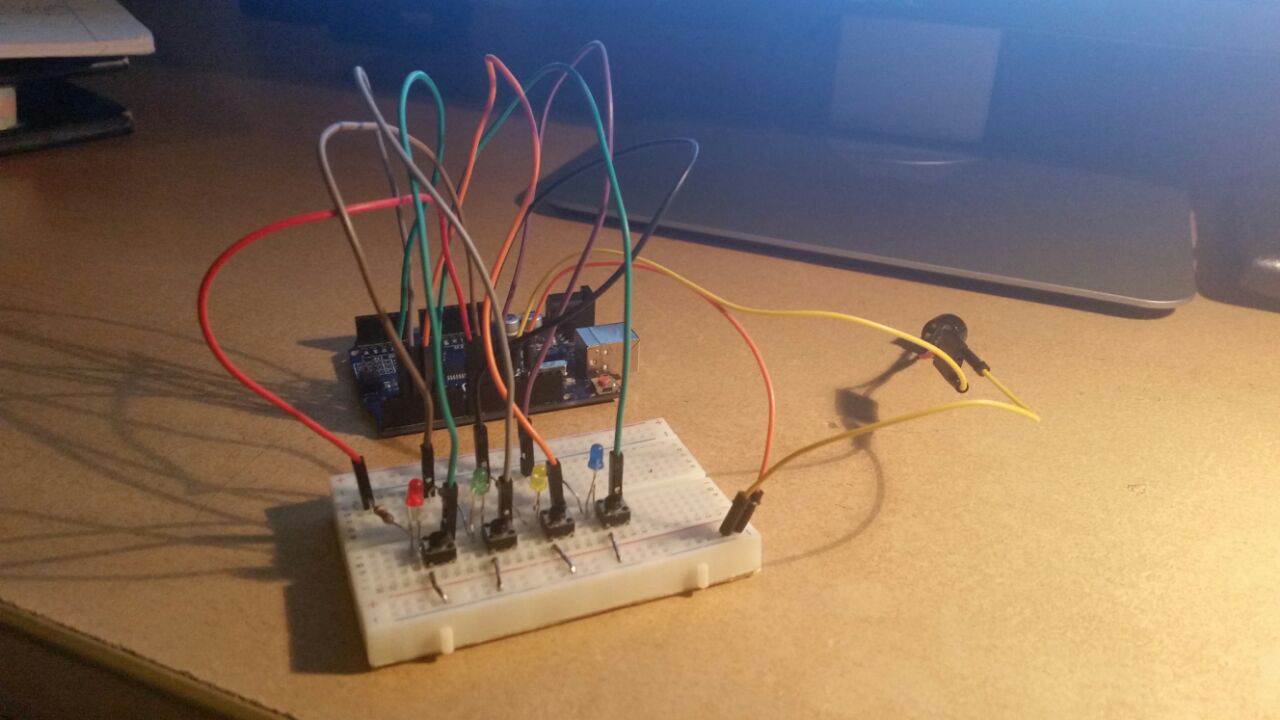
Igra je popularna z otrcoim, in je primerna družinska igra.

**Moj projekt:**

Jaz sem se odločil, da bom za svoj Arduino projekt naredil Simon Says igrco.

Moje komponente sem dobu od sestriča.

Naprava ima štiri gumbe, katerih ima vsak poseben ton, ko pritisne ali aktivira napravo. Sestavljena je iz štirih LED diod do enega ali več gumbov v naključnem vrstnem redu, po katerem mora igralec reprodukcijo tega odloka s pritiskanjem gumbov. Ko se igra napreduje, število gumbov ki jih je treba pritisniti se povečuje.

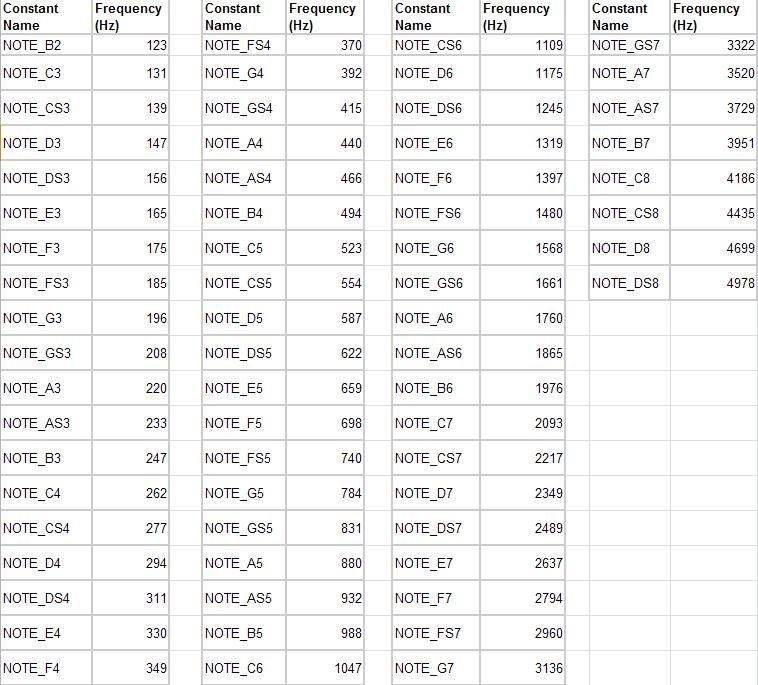


Slika 2: Prikaz mojega projekta

**Moj video:** <https://www.youtube.com/watch?v=f7DdJCIGYNg>

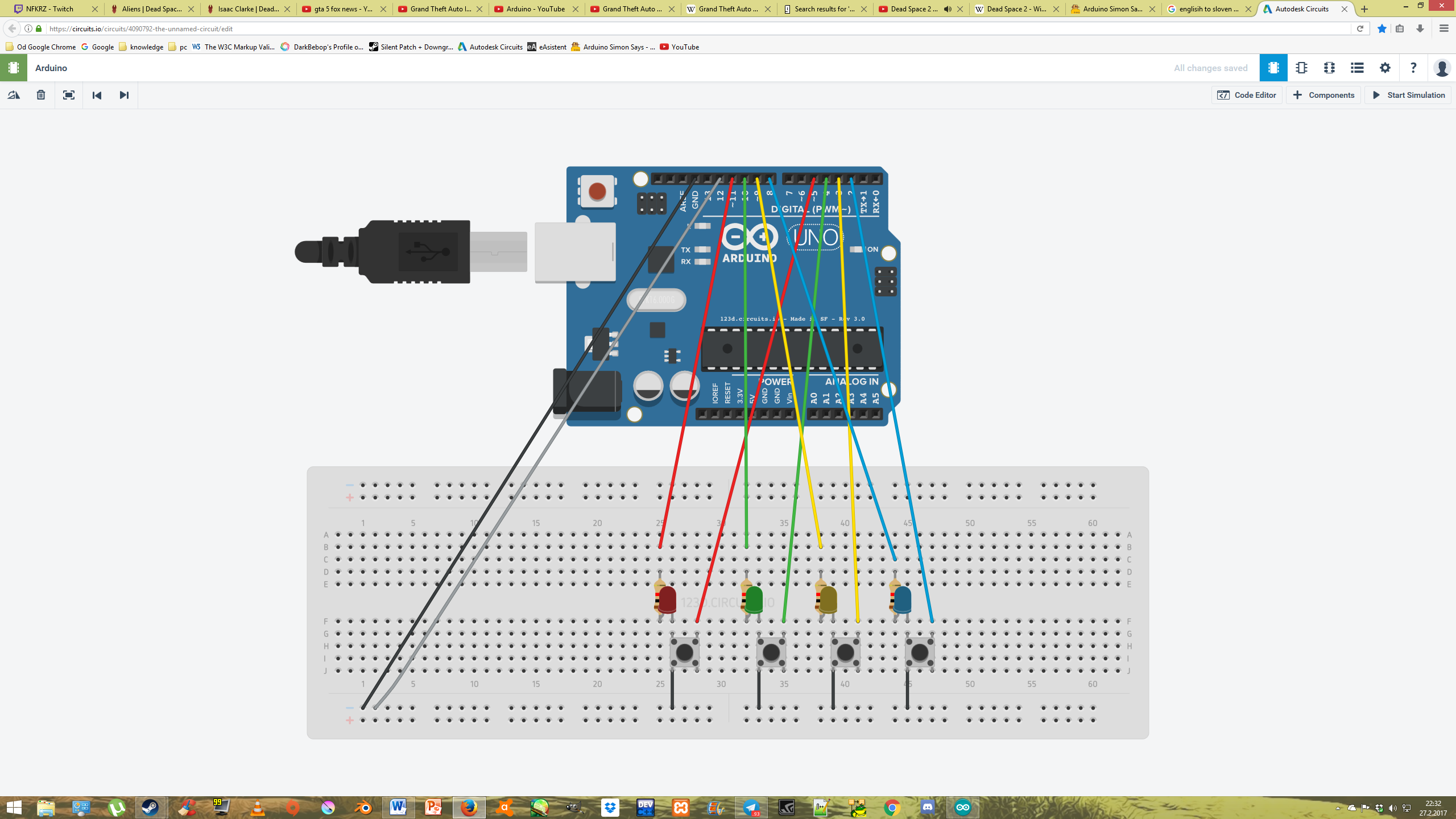
**Kaj sem potreboval?**

* x1 Arduino: <https://www.sparkfun.com/products/11224>
* x1 Bread Board: <https://www.sparkfun.com/products/14083>
* x10 žic: <https://www.sparkfun.com/products/11026>
* x4 različno pobarvanih LED diod: <https://www.sparkfun.com/products/12062>
* x4 100 Ohmske upore: [https://www.sparkfun.com/products/11507](https://www.sparkfun.com/products/11507?_ga=1.167830503.736224130.1488232087)
* x4 gumbe: <https://www.sparkfun.com/products/9190>
* x1 mali zvočnik: <https://www.sparkfun.com/products/7950>
* Program Arduino: <https://www.arduino.cc/>
* Knjižnica Tone: <https://cdn.instructables.com/ORIG/F0C/SG3D/GYUY0NUA/F0CSG3DGYUY0NUA.zip>



Slika 3: Sketch mod za knjižnico Tone.

**Sestava**



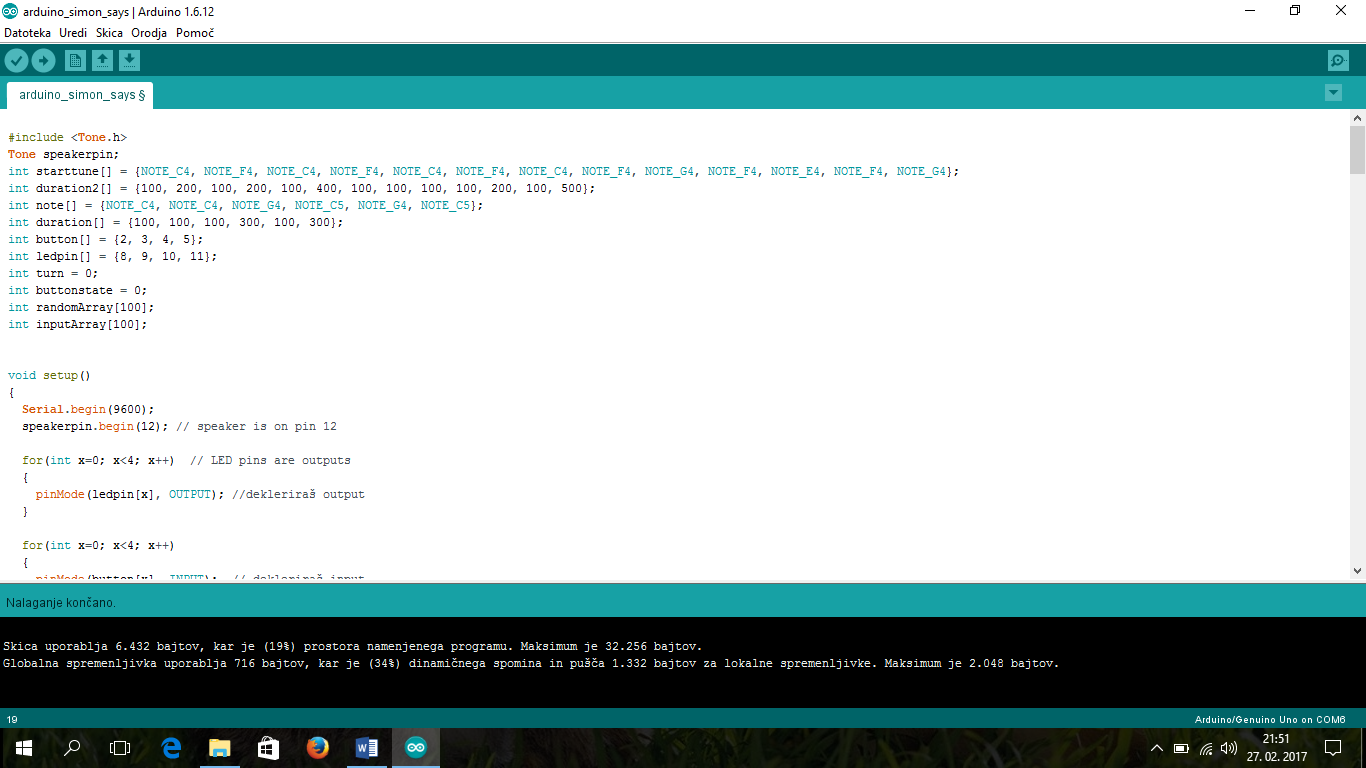
Slika 4: Prikaz projekta (minus zvočnik) v Autodesk circuits <https://circuits.io/>

* Gumb 1 na pin 2 z LED (rdečo) na pin 8
* Gumb 2 na pin 3 z LED (zeleno) na pin 9
* Gumb 3 na pin 4 z LED (rumeno) na pin 10
* Gumb 4 na pin 5 z LED (modro) na pin 11
* Zvočnik na pin 12

**Opis kode:**

**Poimenovanje:**

Poimenujemo zvočnik, različne intižerje za pesmi, LED diode, gumbe in števec.



**#include <Tone.h>** vključuje knjižnico "Tone" v C:\Program Files (x86)\Arduino\libraries.

**Tone speakerpin;** poimenovali smo zvočnik in bo zdej uporabljal knjižnico "Tone".

**int starttune[ ] = {NOTE\_C4, NOTE\_F4… };** to je začetna pesem, ki jo Arduino zaigra ko se zažene. Za starttune smo izbrali 13 ton (0-12), ki jih bo zvočnik izpustil.

**int duration2[ ] = {100, 200… };** to je čas vsakega tona za starttune, 1000 = 1 sekunda in tako naprej.

**int note[ ] = {NOTE\_C4, NOTE\_C4… };** to je pesem, ko zaigra ko vam uspe taprav gumb pretisnat.

**int duration[ ] = {100, 100… };** to je čas vsakega tona za note, 1000 = 1 sekunda, 100 = 0.1 sekunda in tako naprej.

**int button[ ] = {2, 3, 4, 5};** input pini za 4 gumbe.

**int ledpin[ ] = {8, 9, 10, 11};** output pini za 4 LED diode.

**int turn = 0;** števec, ki šteje kolikokrat vam je uspelo uganit.

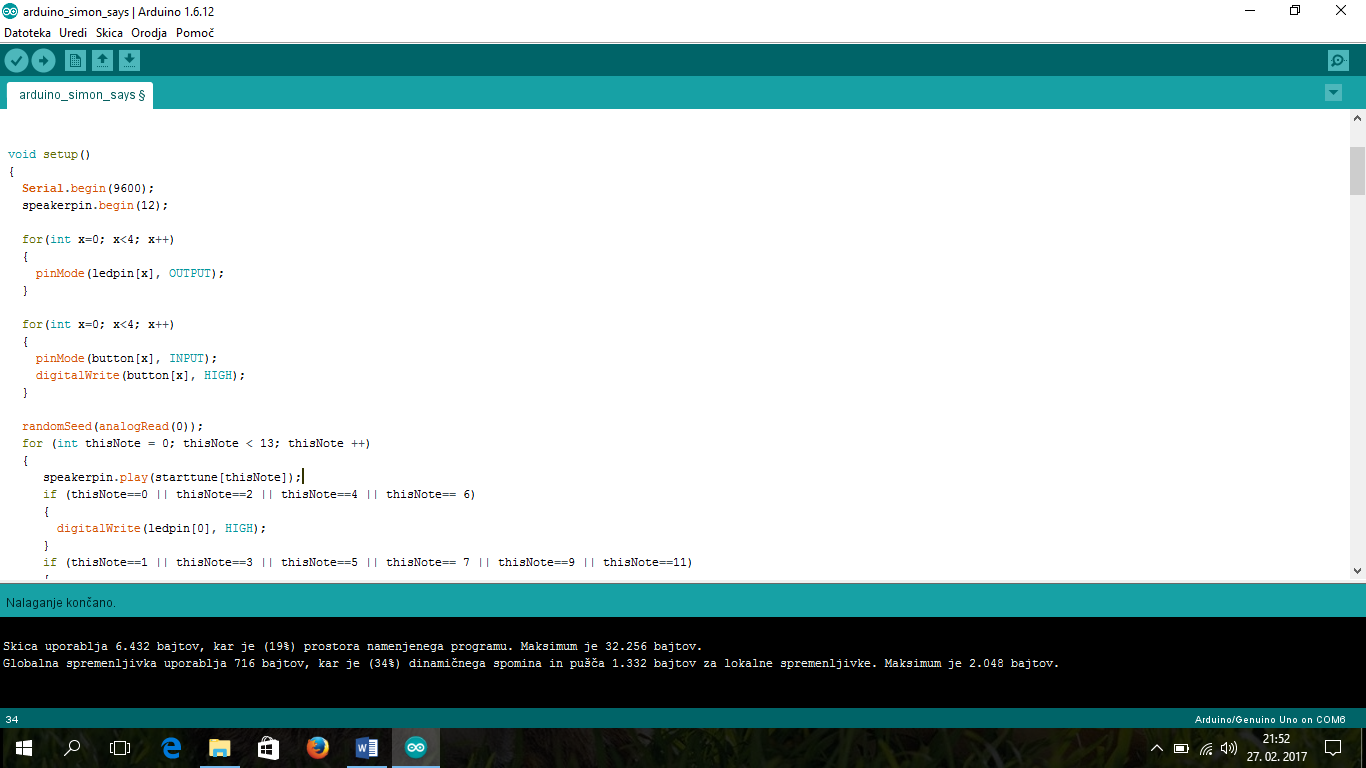
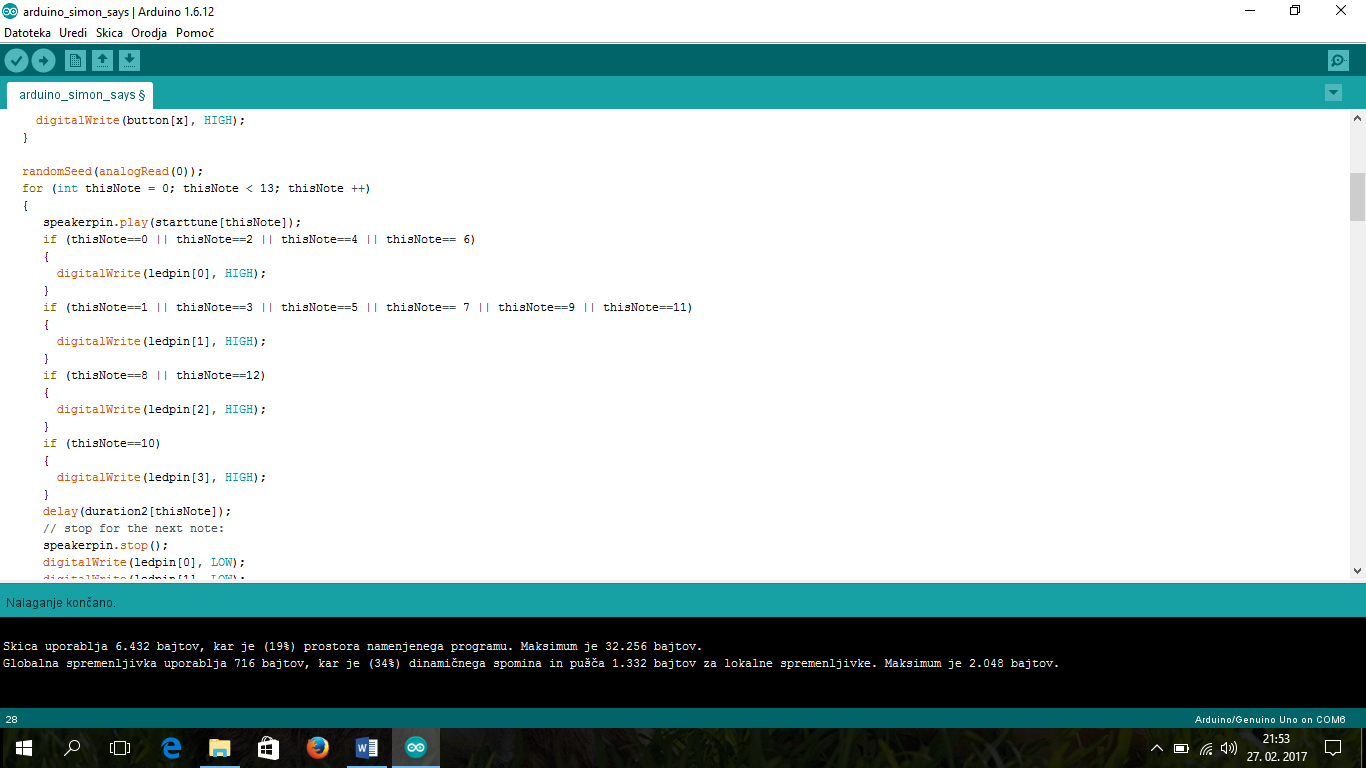
**int buttonstate = 0;** preverja stanje gumba, če je uporabnik ga pritisnal.

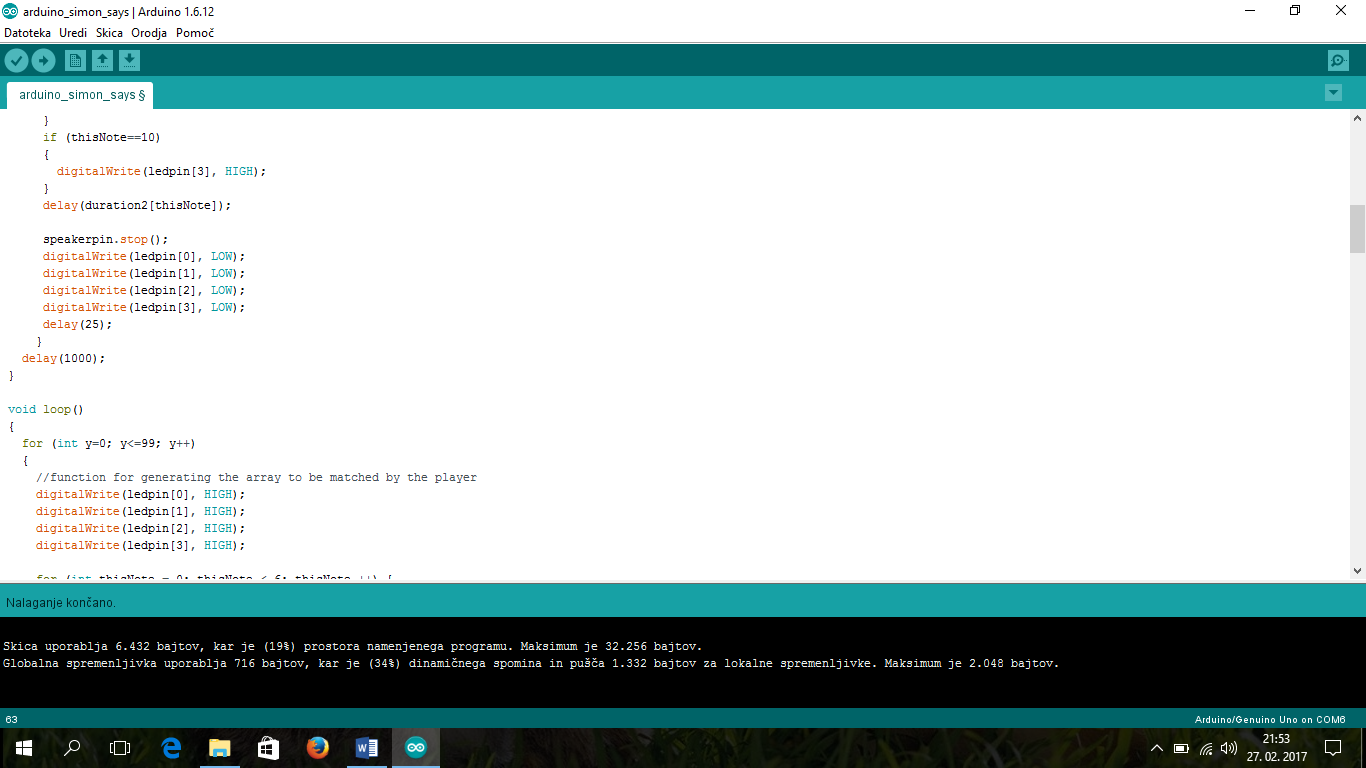
**int randomArray[100];** izmislil si bo 100 naključnih rezultatov, ki jih more uporabnik ugotovit.

**int inputArray[100];** hrani okrog 100 naključnih inputov.

**Void setup:**

Dekleriramo LED diode (da so outputi), gumbe (da so inputi) in da bo igrala začetno pesem.





**void setup()**

{

**Serial.begin(9600);** nastavimo 9600 bitov na sekundo.

**speakerpin.begin(12);** zvočnik smo nastavli na pin 12.

**for(*int* x=0; x<4; x++)** dekleriramo vse 4 LED diode (0-3) na OUTPUT.

{

**pinMode(ledpin[x], OUTPUT);** deklerira vsako LED diodo.

}

**for(*int* x=0; x<4; x++)** dekleriramo vseh 4 gumbov (0-3) na INPUT.

{

**pinMode(button[x], INPUT);** deklerira vsakega gumba.

**digitalWrite(button[x], HIGH);** da napetost 5V na vsakega gumba.

}

**randomSeed(analogRead(0));** dodal sem, da bo ustvarila večjo naključnosti, da je bolj random, z randomArray za izhode..

**for (int thisNote = 0; thisNote < 13; thisNote ++)** : tule ti zaigra začetno pesem, "starttune" (0-13)

{

**speakerpin.play(starttune[thisNote]);** začne igrat tone iz starttune.

**if (thisNote==0 || thisNote==2 || thisNote==4 || thisNote== 6)** če je 0,2,4 ali 6 se vklopi rdeča LED dioda.

{

**digitalWrite(ledpin[0], HIGH);** rdeča LED dioda bo zastevtla, ko bo zvočnik izpustil tone.

}

**if (thisNote==1 || thisNote==3 || thisNote==5 || thisNote== 7 || thisNote==9 || thisNote==11)** če je 1,3,5 in 7 se vklopi zelena LED dioda.

{

**digitalWrite(ledpin[1], HIGH);** zelena LED dioda se bo zastevtla ko bo zvočnik izpustil tone.

}

**if (thisNote==8 || thisNote==12)** če je 8 ali 12 se vklopi rumena LED dioda.

{

**digitalWrite(ledpin[2], HIGH);** : rumena LED dioda se bo zastevtla, ko bo zvočnik izpustil tone.

}

**if (thisNote==10)** če je 10 se vklopi modra LED dioda.

{

**digitalWrite(ledpin[3], HIGH);** modra LED dioda se bo zastevtla, ko bo zvočnik izpustil tone.

}

**delay(duration2[thisNote]);** čas čakanja za naslednjo tono.

**speakerpin.stop();** zvočnik se ugasne.

**digitalWrite(ledpin[0], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[1], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[2], LOW);** LED dioda se ugasne.**.**

**digitalWrite(ledpin[3], LOW);** LED dioda se ugasne.

**delay(25);**

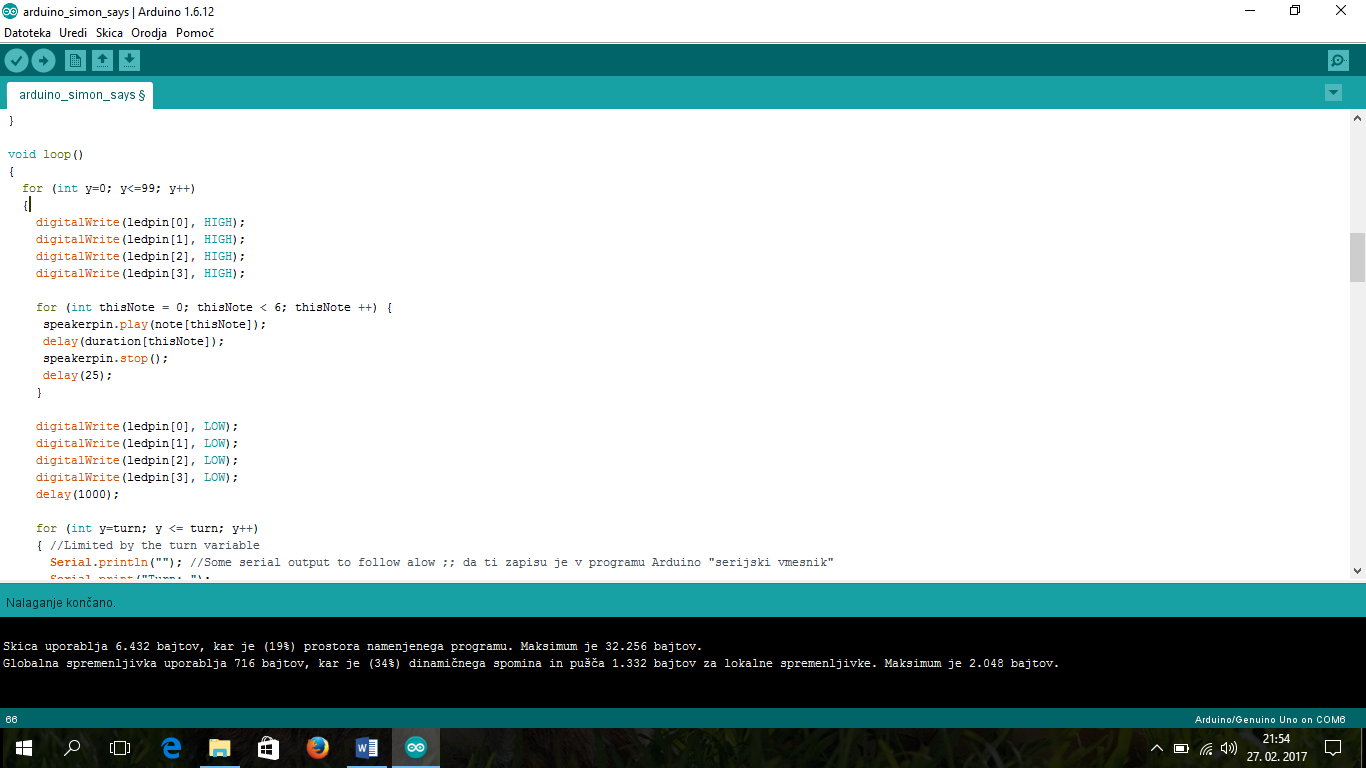
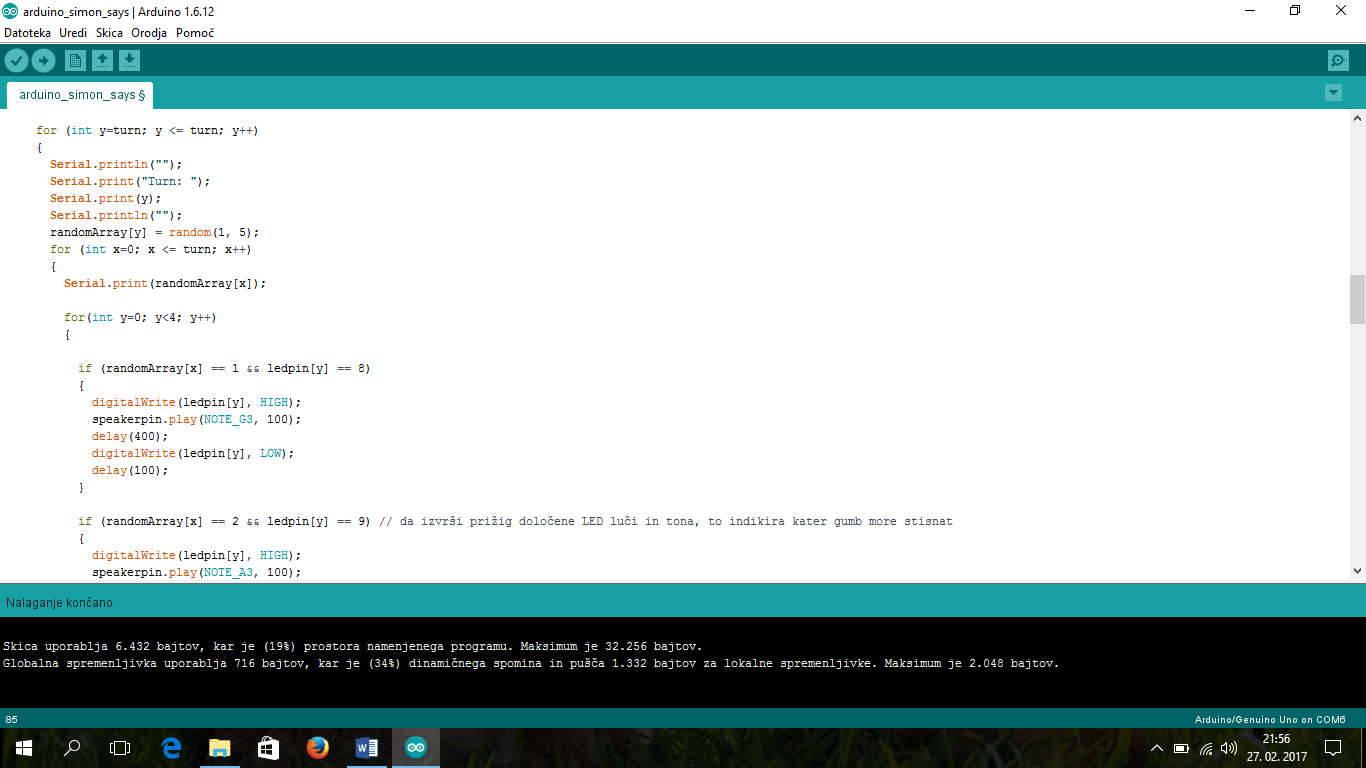
}

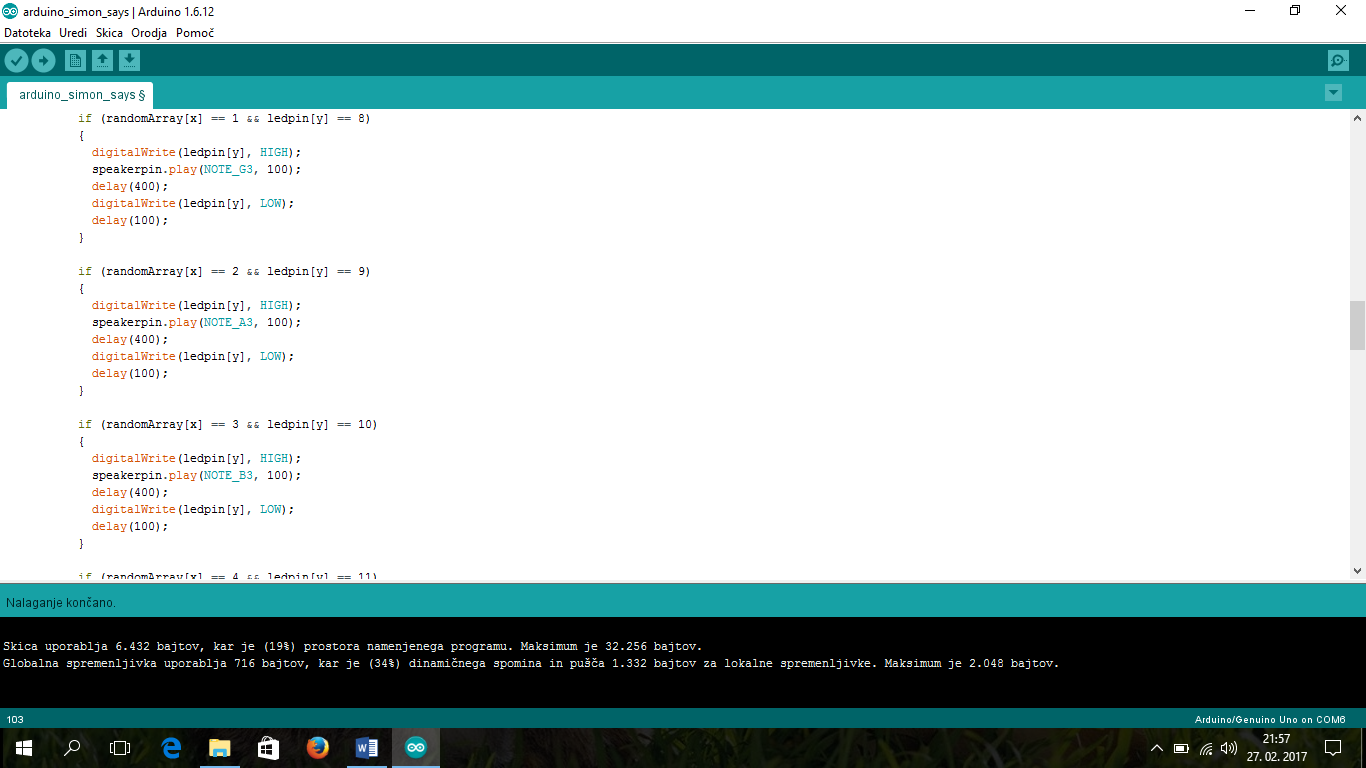
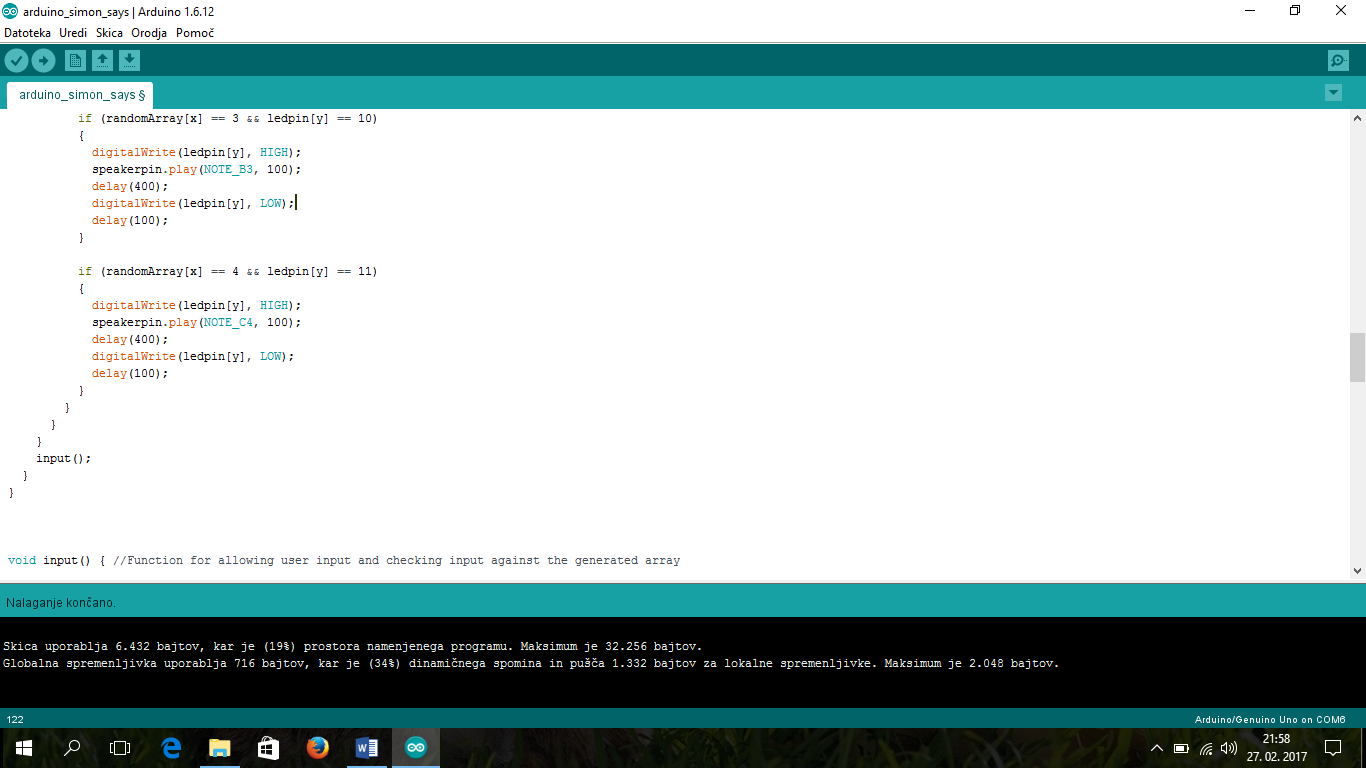
**delay(1000);** : čaka 1 sekundo, in nato bo šlo na naslednjo tono.

}

**Void loop:**

Arduino izbere naključno število od (1-4) in se določena LED dioda zasveti in si jo mora uporabnik zapomnit.





**void loop()**

{

**for (int y=0; y<=99; y++)** da lahko uporabnik igra do 100 (0-99).

{

**digitalWrite(ledpin[0], HIGH);** LED dioda se prežge.

**digitalWrite(ledpin[1], HIGH);** LED dioda se prežge.

**digitalWrite(ledpin[2], HIGH);** LED dioda se prežge.

**digitalWrite(ledpin[3], HIGH);** LED dioda se prežge.

**for (int thisNote = 0; thisNote < 6; thisNote ++)** tule ti zaigra začetno pesem, "note" (0-5)

{

**speakerpin.play(note[thisNote]);** začne igrat tone iz note.

**delay(duration[thisNote]);** čas čakanja za naslednjo tono.

**speakerpin.stop();** zvočnik se ugasne.

**delay(25);**

}

**digitalWrite(ledpin[0], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[1], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[2], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[3], LOW);** LED dioda se ugasne.

**delay(1000);**

**for (int y=turn; y <= turn; y++)**

{

**Serial.println("");** nova vrstica, ki se zapisuje v serijski vmesnik.

**Serial.print("Turn: ");** natisne števec Turn v serijski vmesnik.

**Serial.print(y);** natisne vrednost y v serijski vmesnik.

**Serial.println("");** nova vrstica v serijski vmesniku.

**randomArray[y] = random(1, 5);** določanje naključnega števila (1-4).

**for (int x=0; x <= turn; x++)**

{

**Serial.print(randomArray[x]);**

**for(int y=0; y<4; y++)**

{

**if (randomArray[x] == 1 && ledpin[y] == 8)** da izvrši prižig določene LED luči in tona, to indikira kater gumb more stisnat.

{

**digitalWrite(ledpin[y], HIGH);** rdeča LED dioda se bo zastevtla.

**speakerpin.play(NOTE\_G3, 100);** zaigra tono, ko se LED dioda prežge.

**delay(400);** čas svetlobe.

**digitalWrite(ledpin[y], LOW);** LED dioda se ugasne

**delay(100);**

}

**if (randomArray[x] == 2 && ledpin[y] == 9)** da izvrši prižig določene LED luči in tona, to indikira kater gumb more stisnat.

{

**digitalWrite(ledpin[y], HIGH);** zelena LED dioda se bo zastevtla.

**speakerpin.play(NOTE\_A3, 100);** zaigra tono, ko se LED dioda prežge.

**delay(400);** čas svetlobe.

**digitalWrite(ledpin[y], LOW);** LED dioda se ugasne.

**delay(100);**

}

**if (randomArray[x] == 3 && ledpin[y] == 10)** da izvrši prižig določene LED luči in tona, to indikira kater gumb more stisnat.

{

**digitalWrite(ledpin[y], HIGH);** rumena LED dioda se bo zastevtla.

**speakerpin.play(NOTE\_B3, 100);** zaigra tono, ko se LED dioda prežge.

**delay(400);** čas svetlobe.

**digitalWrite(ledpin[y], LOW);** LED dioda se ugasne.

**delay(100);**

}

**if (randomArray[x] == 4 && ledpin[y] == 11)** da izvrši prižig določene LED luči in tona, to indikira kater gumb more stisnat.

{

**digitalWrite(ledpin[y], HIGH);** modra LED dioda se bo zastevtla.

**speakerpin.play(NOTE\_C4, 100);** zaigra tono, ko se LED dioda prežge.

**delay(400);** čas svetlobe.

**digitalWrite(ledpin[y], LOW);** LED dioda se ugasne.

**delay(100);**

}

}

}

}

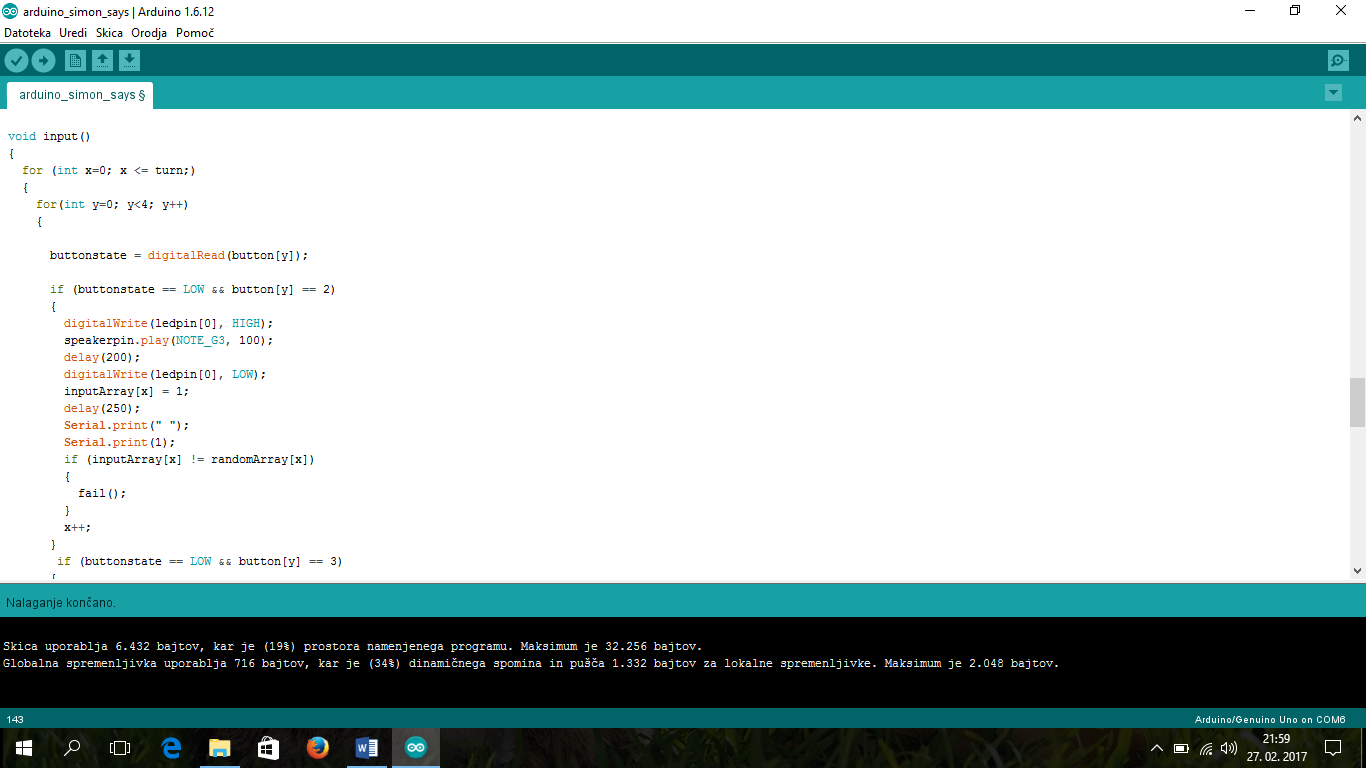
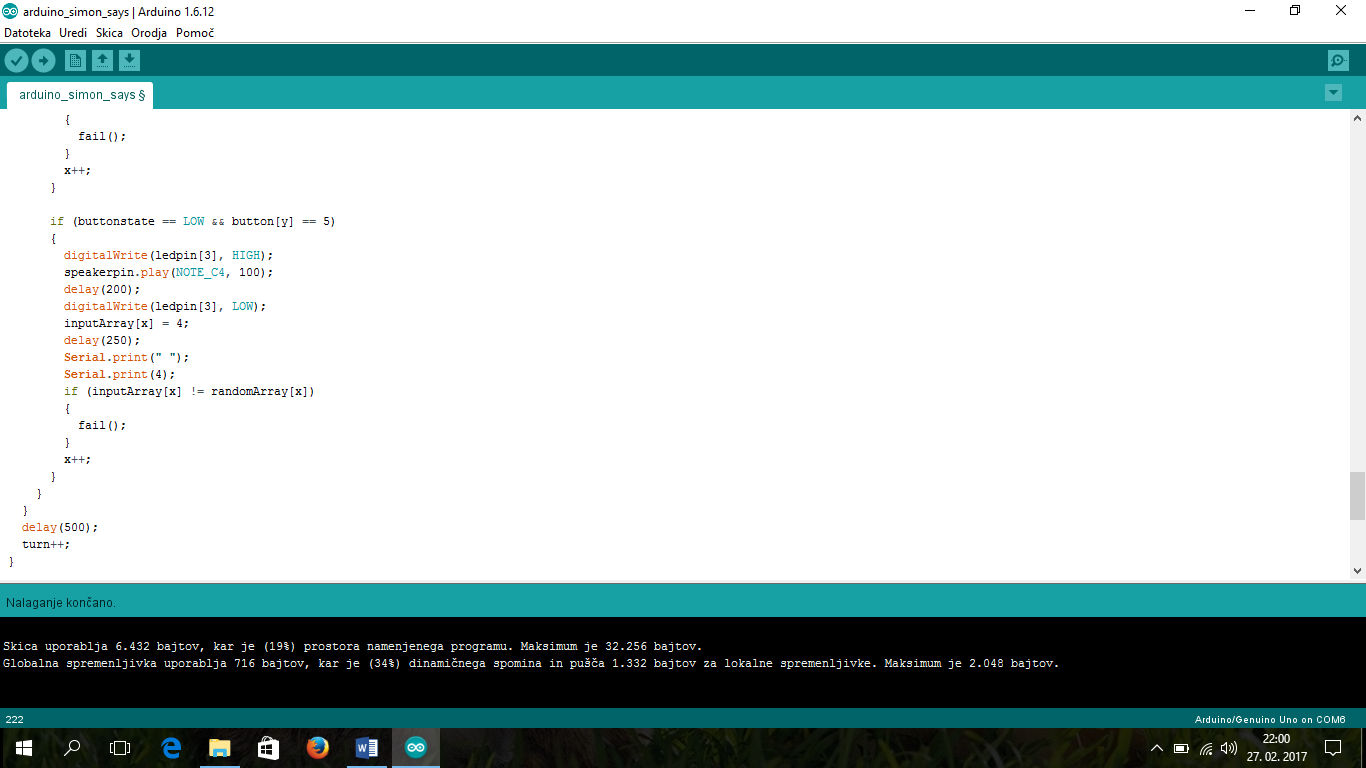
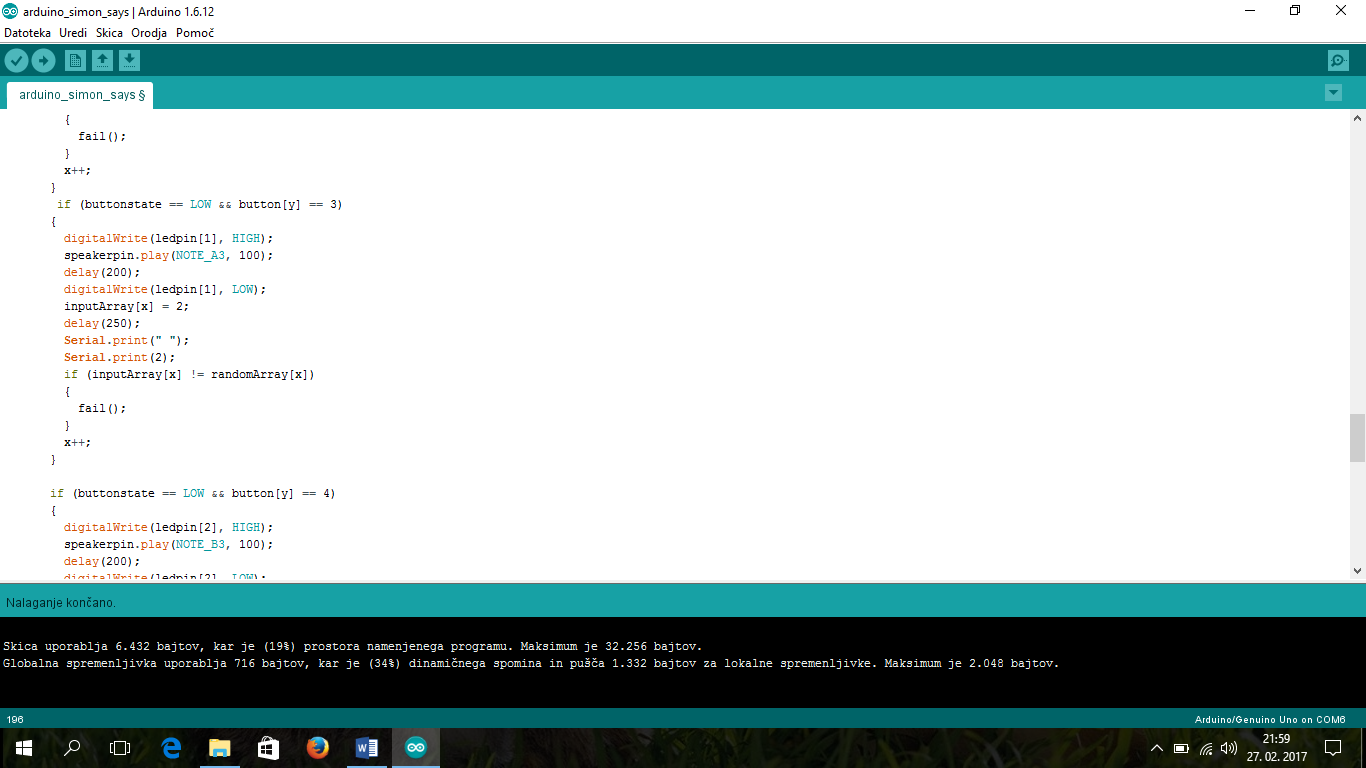
**input();** uporablja, naslednjo kodo.

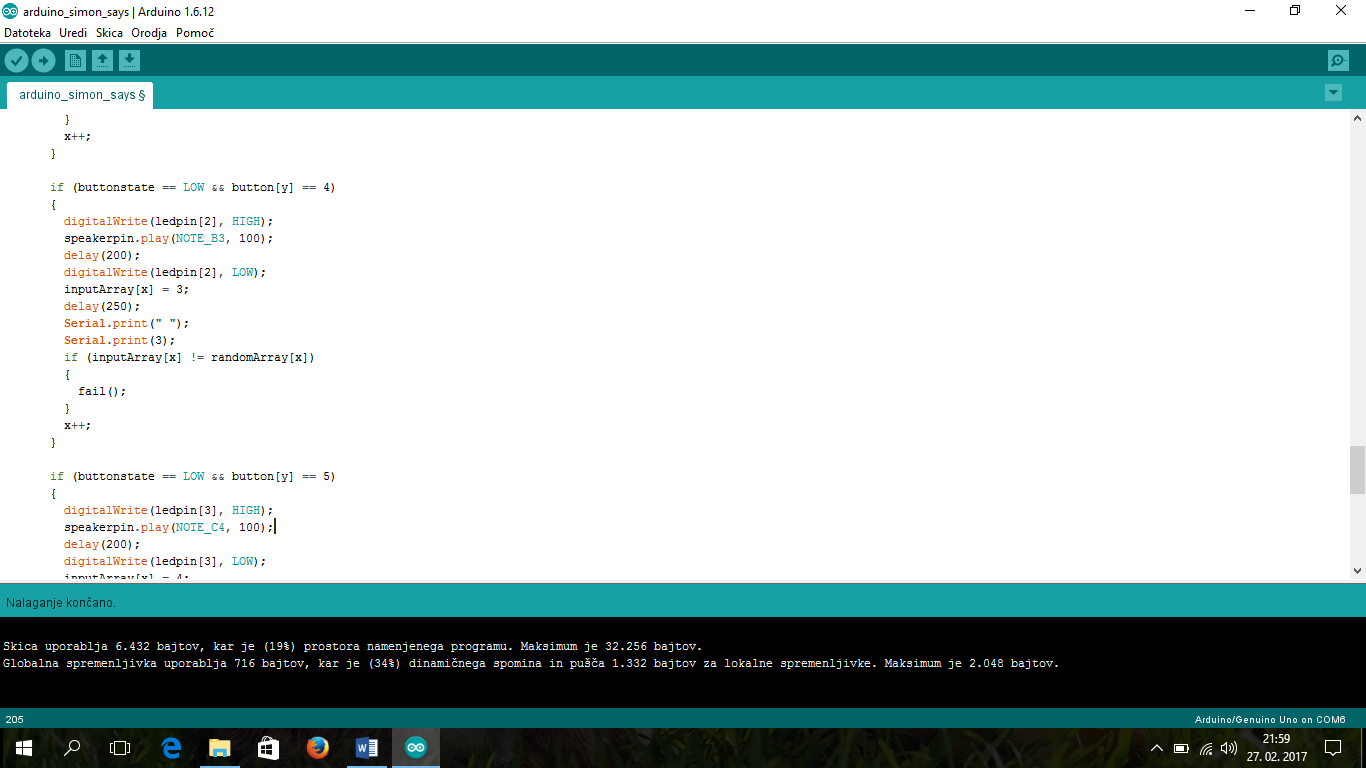
}

}

**Void input:**

Uporabnik more pretisnat gumb, kjer se je zasvetla LED dioda, če mu uspe gre naprej





**void input()** Funkcija omogoča uporabnikov vnos in preverjanje vhod proti ustvarjeni niz.

{

**for (int x=0; x <= turn;)** izjava nadzorovana s štetjem turn.

{

**for(int y=0; y<4; y++)** za vsak gumb

{

**buttonstate = digitalRead(button[y]);** preveri, kater gumb je uporabnik pretisnal.

**if (buttonstate == LOW && button[y] == 2)** potem ko stisneš naj se prežge luč in se izvrši ton, ter izpiše vmesnik kateri gumb si stisnu

{

**digitalWrite(ledpin[0], HIGH);** rdeča LED dioda zasveti.

**speakerpin.play(NOTE\_G3, 100);** izpusti Tono.

**delay(200);** čas čakanja.

**digitalWrite(ledpin[0], LOW);** LED dioda se ugasne.

**inputArray[x] = 1;** nastavimo na 1, da preveri v naslednjem if stavku.

**delay(250);** čas čakanja.

**Serial.print(" ");** nova vrstica v serijski vmesniku.

**Serial.print(1);** vpiše 1 v Serijski vmesnik.

**if (inputArray[x] != randomArray[x])** pregleda vrednost, ki jo je uporabnik vnesu.

{

**fail();** če uporabnik napačen gumb pretisnal, gre na nasledno kodo.

}

**x++;** x se poveča za 1.

}

**if (buttonstate == LOW && button[y] == 3)** potem ko stisneš se prežge LED dioda in se izvrši ton, ter izpiše vmesnik kateri gumb si stisnu.

{

**digitalWrite(ledpin[1], HIGH);** zelena LED dioda zasveti.

**speakerpin.play(NOTE\_A3, 100);** izpusti Tono.

**delay(200);** čas čakanja.

**digitalWrite(ledpin[1], LOW);** LED dioda se ugasne.

**inputArray[x] = 2;** nastavimo na 2, da preveri v naslednjem if stavku.

**delay(250);** čas čakanja.

**Serial.print(" ");** nova vrstica v serijski vmesniku.

**Serial.print(2);** vpiše 2 v Serijski vmesnik.

**if (inputArray[x] != randomArray[x])** pregleda vrednost, ki jo je uporabnik vnesu.

{

**fail();**

}

**x++;** x se poveča za 1.

}

**if (buttonstate == LOW && button[y] == 4)** potem ko stisneš se prežge LED dioda in se izvrši ton, ter izpiše vmesnik kateri gumb si stisnu.

{

**digitalWrite(ledpin[2], HIGH);** rumena LED dioda zasveti.

**speakerpin.play(NOTE\_B3, 100);** izpusti Tono.

**delay(200);** čas čakanja.

**digitalWrite(ledpin[2], LOW);** LED dioda se ugasne.

**inputArray[x] = 3;** nastavimo na 3, da preveri v naslednjem if stavku.

**delay(250);** čas čakanja.

**Serial.print(" ");** nova vrstica v serijski vmesniku.

**Serial.print(3);** vpiše 3 v Serijski vmesnik.

**if (inputArray[x] != randomArray[x])** pregleda vrednost, ki jo je uporabnik vnesu.

{

**fail();** če uporabnik napačen gumb pretisne, gre na nasledno kodo.

}

**x++;** x se poveča za 1.

}

**if (buttonstate == LOW && button[y] == 5**) potem ko stisneš se prežge LED dioda in se izvrši ton, ter izpiše vmesnik kateri gumb si stisnu.

{

**digitalWrite(ledpin[3], HIGH);** modra LED dioda zasveti.

**speakerpin.play(NOTE\_C4, 100);** izpusti Tono.

**delay(200);** čas čakanja.

**digitalWrite(ledpin[3], LOW);** LED dioda se ugasne.

**inputArray[x] = 4;** nastavimo na 4, da preveri v naslednjem if stavku.

**delay(250);** čas čakanja.

**Serial.print(" ");** nova vrstica v serijski vmesniku.

**Serial.print(4);** vpiše 4 v Serijski vmesnik.

**if (inputArray[x] != randomArray[x])** pregleda vrednost, ki jo je uporabnik vnesu.

{

**fail();** če uporabnik napačen gumb pretisne, gre na nasledno kodo.

}

**x++;** x se poveča za 1.

}

}

}

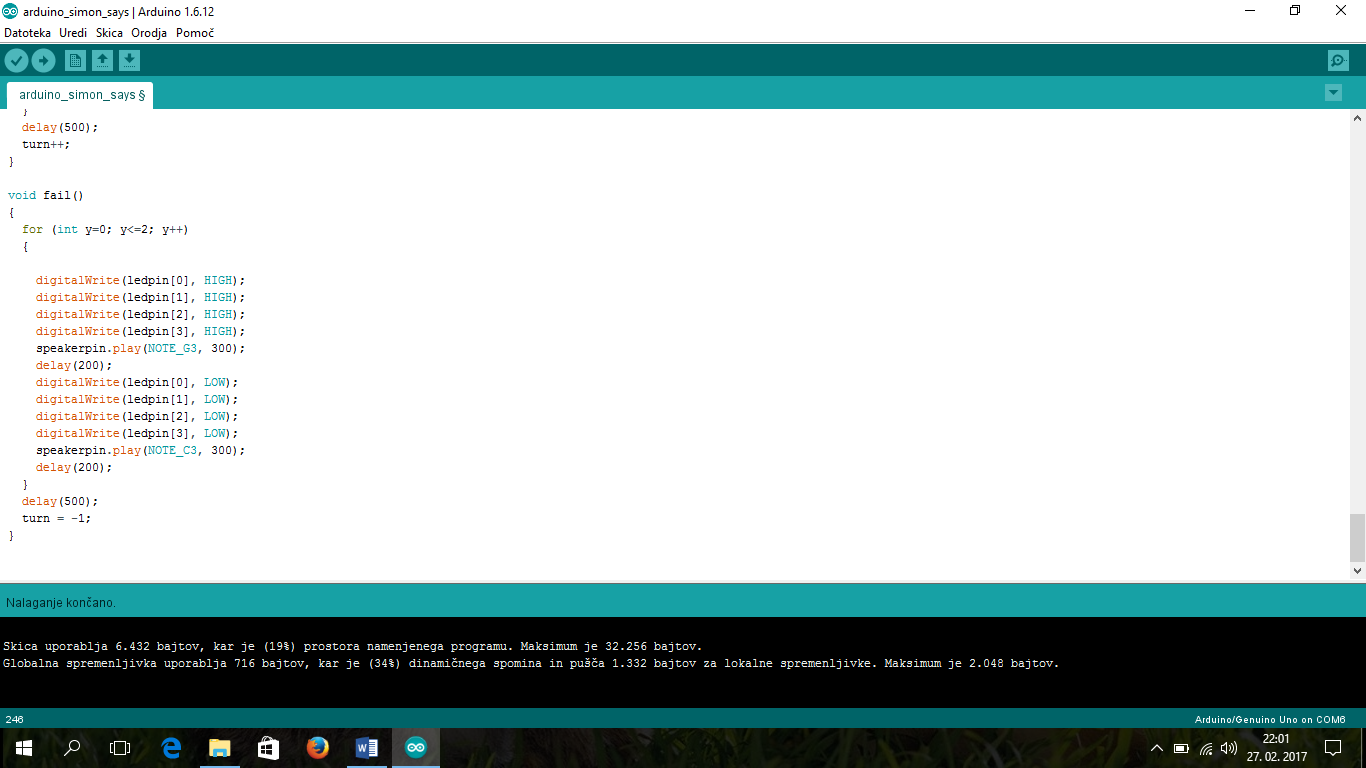
**delay(500);**

**turn++;** poveča Turn števec za 1, preden znova zaženete izhodno funkcijo.

}

**Void fail:**

Če uporabnik zgrši se števec resetira in gre od začetka



**void fail()** funkcija uporablja, če igralec ne ujemajo zaporedje

{

**for (int y=0; y<=2; y++)** utripajo LED diode za neuspeh

{

**digitalWrite(ledpin[0], HIGH);** LED dioda se prežege.

**digitalWrite(ledpin[1], HIGH);** LED dioda se prežege.

**digitalWrite(ledpin[2], HIGH);** LED dioda se prežege.

**digitalWrite(ledpin[3], HIGH);** LED dioda se prežege.

**speakerpin.play(NOTE\_G3, 300);** izpusti tono.

**delay(200);** čas čakanja

**digitalWrite(ledpin[0], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[1], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[2], LOW);** LED dioda se ugasne.

**digitalWrite(ledpin[3], LOW);** LED dioda se ugasne.

**speakerpin.play(NOTE\_C3, 300);** izpusti tono.

**delay(200);** čas čakanja

}

**delay(500);**

**turn = -1;** resetira turn števec in greš nazaj na začetek.

}

**Viri:**

<https://en.wikipedia.org/wiki/Simon_(game)>

<http://www.instructables.com/id/Arduino-Simon-Says>

<https://learn.sparkfun.com/tutorials/sik-experiment-guide-for-arduino---v32/experiment-16-simon-says>

<https://create.arduino.cc/projecthub/Arduino_Scuola/a-simple-simon-says-game-25ba99>

<https://www.youtube.com/watch?v=FUtQzdkgMhE>

**Domen Stropnik**

**3.TRB**