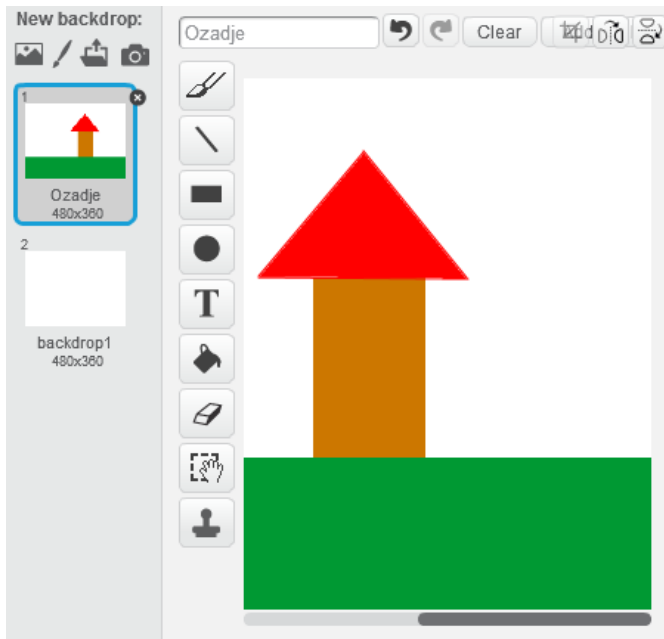


NALOGE Z OZADJEM

Pri naslednjih nalogah se bomo ukvarjali z barvami. Za ta namen boš narisal različna ozadja. Začni s prvim ozadjem. Klikni na ikono Ozadje (Stage). Izberi zavihek Ozadja in klikni gumb Uredi.

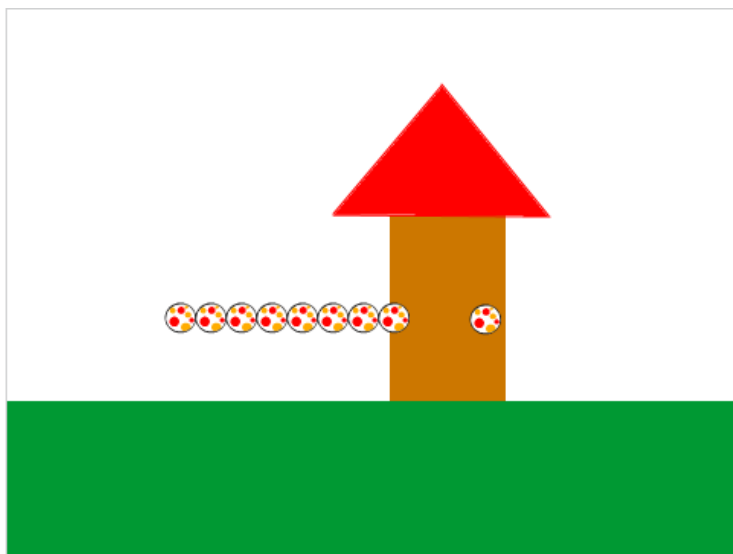


V tej nalogi bomo figuro mačka zbrisali in uvozili novo s klikom na gumb Izberi nov sprite iz datoteke.



V mapi Stvari (Things) izberi figuro žogo (beachball1).

Napiši program, ki žogo 10-krat premakne za 30 korakov v desno, odtis pa pusti le, če je barva ozadja bela.



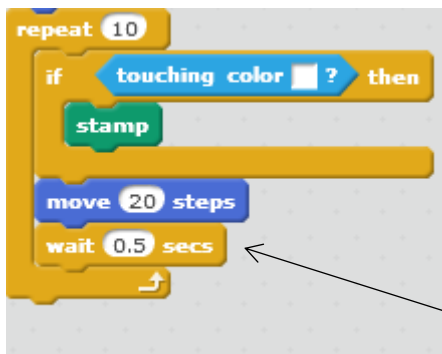
V skupini Zaznavanje (Sensing) poišči ukaz, ki prepozna barve na odru: in s pomočjo kapalke določi ustrezno barvo.



Sestavimo ukazni blok, ki naredi 10 odtisov žog.



Ker želiš odtis žoge le na belem ozadju, dodaj še pogoj: če se žoga dotika bele barve, pusti odtis. Premik žoge naj se zgodi v vsakem primeru.

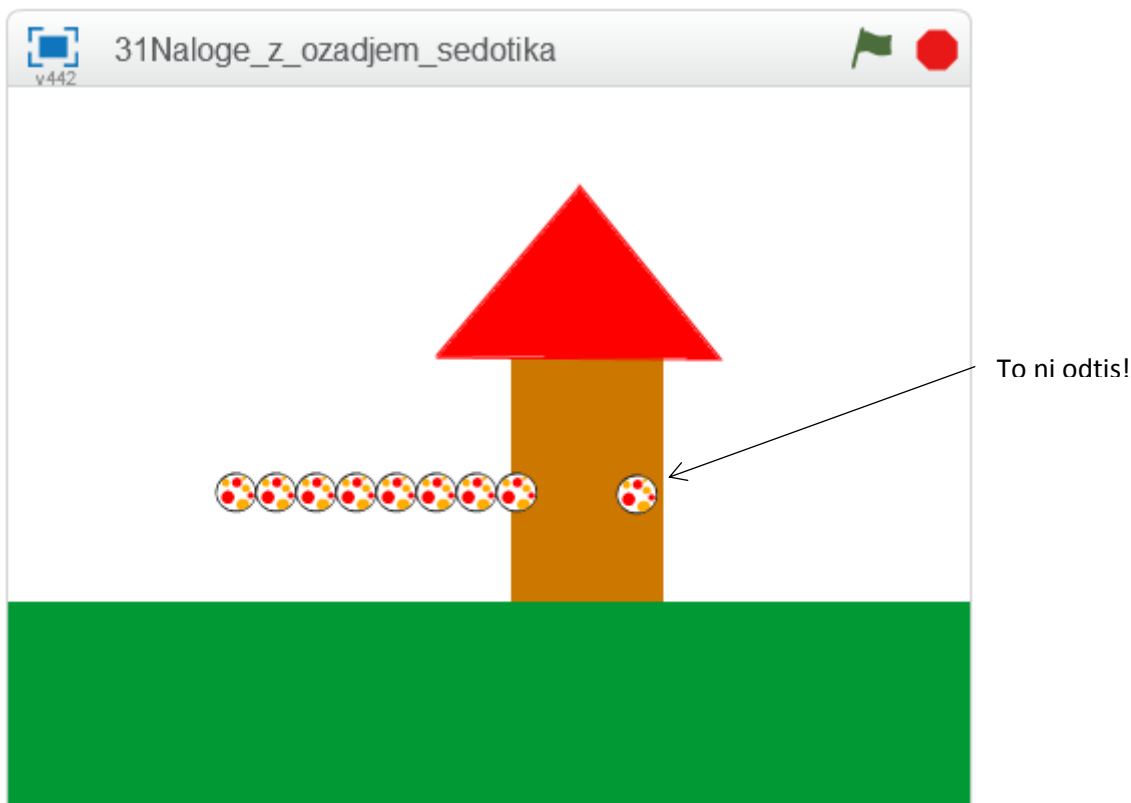


Ukaz dodamo, da upočasnimo risanje.

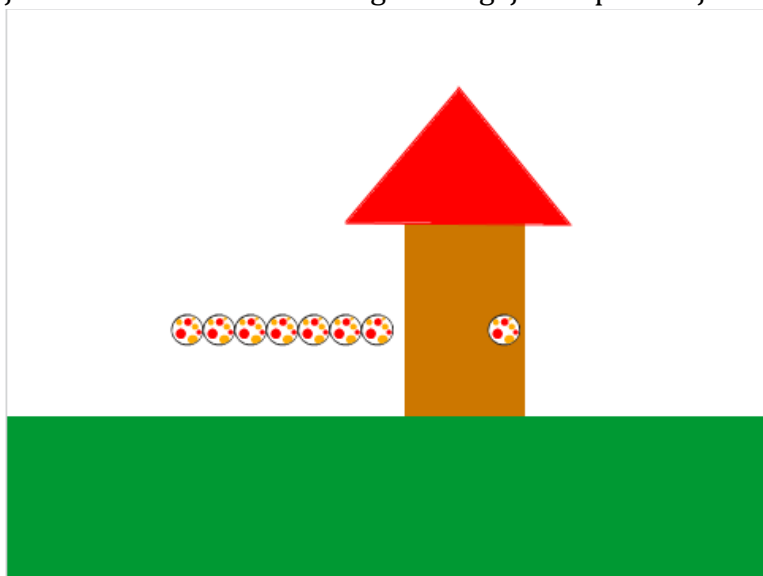
Programu dodamo še začetni ukaz in začetni položaj žoge:



Kot vidiš, žoga pusti odtis tudi tam, kjer se le delno dotika bele barve. To bomo poskušali rešiti v naslednji nalogi. Ne pozabi, da je zadnja žoga v vrsti figura in ne odtis.



Preuredi prejšnji program tako, da žoga 10-krat premakne za 20 korakov, odtis pa pusti le, če je barva v celoti bela. Namig: Pomagaj si z operatorjem .

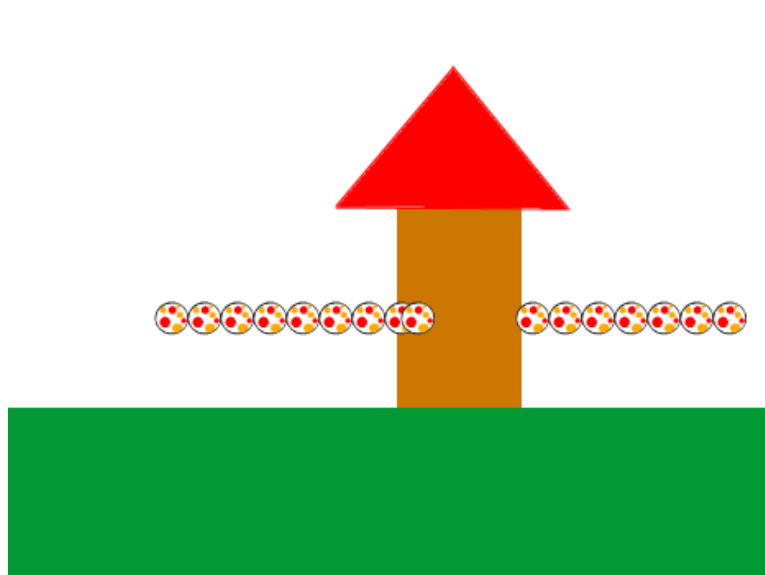


PONOVI DOKLER, ZA VEDNO ČE

Doslej smo se srečevali s problemi, ko je bilo za neko dejanje natančno določeno, kolikokrat naj se ponovi, npr. 3-krat, 5-krat. Pogosto pa ne veš vnaprej, kolikokrat se bo neko dejanje ponovilo. Če na primer želiš priti do konca nogometnega igrišča, ne moreš reči: »Naredi 100 korakov.« Pravilno bi rekli: »Hodi toliko časa, dokler ne prideš do konca igrišča.«

Poglejmo si primer:

Program bomo napisali tako, da bo žoga puščala odtise na beli površini, dokler ne pride do konca odra. Ko pride do konca odra, se postavi točno na sredino.



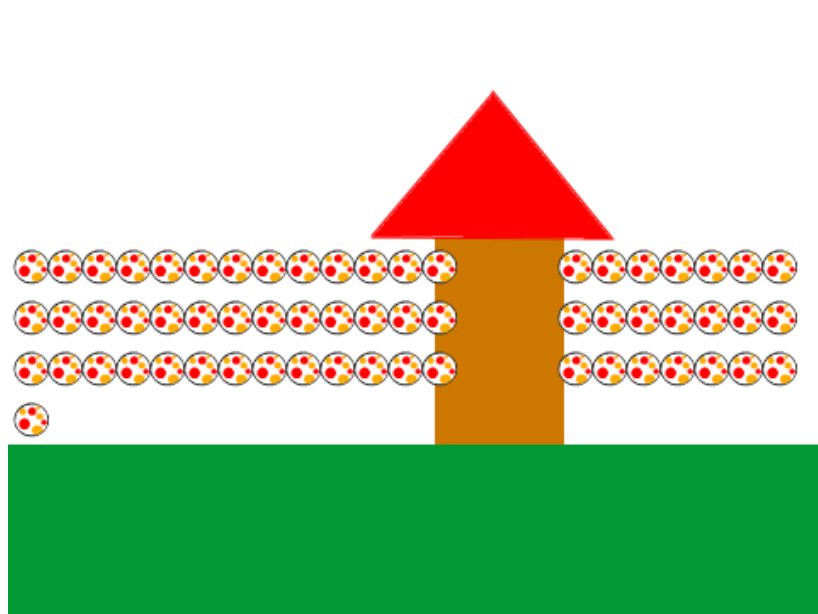
Nalogo lahko rešimo na dva načina: z uporabo ukaza ponovi dokler

Rešitev:



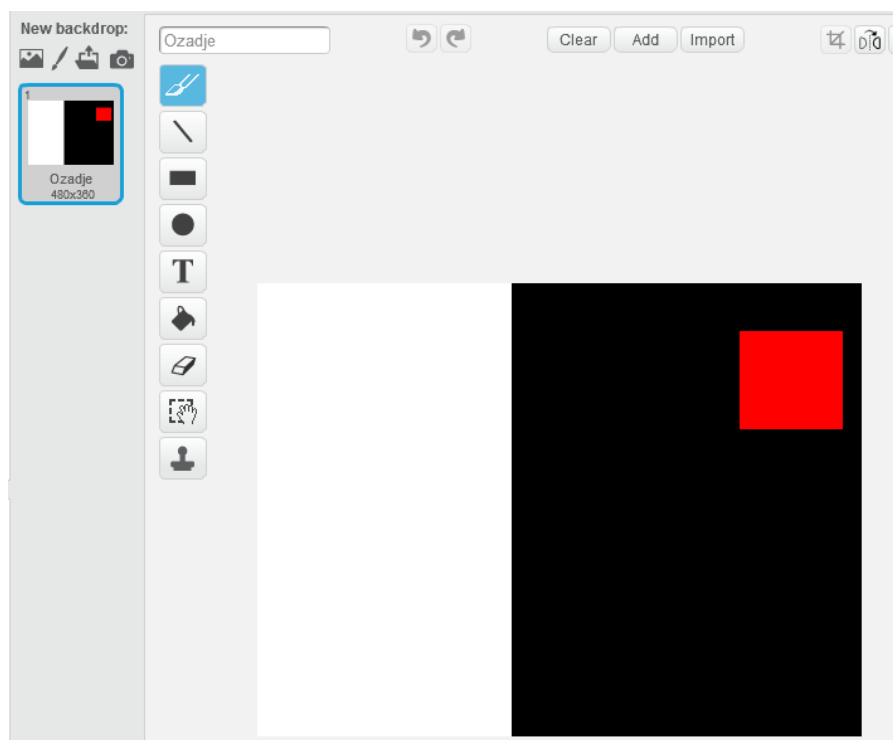
Zanka se bo izvajala toliko časa, dokler žoga ne bo prišla do roba platna. Torej, če na vprašanje Ali se dotika roba?, odgovorimo z da, se izvajanje zanke konča. Program se nato nadaljuje z ukazom pojdi na x: y:

Napiši program , ki s pomočjo zanke nariše tri vrste žog na beli podlagi od začetka do konca zaslona.



ČE, DRUGAČE PA

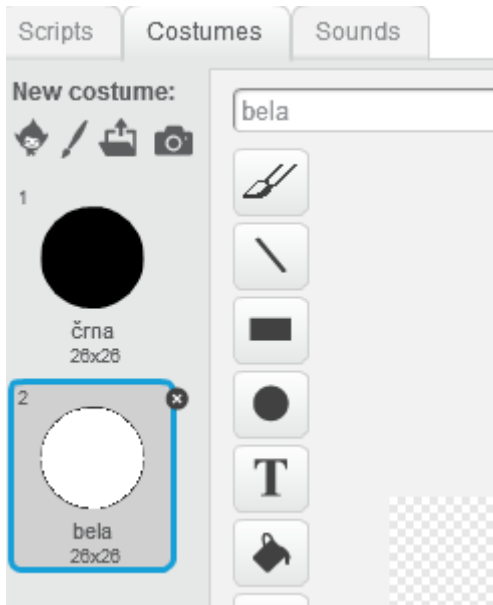
Odpri nov projekt in nariši novo ozadje s tremi različnimi barvami.



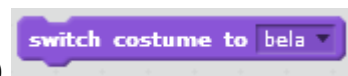
Nariši novo figuro, majhno črno piko, in jo poimenuj Pika. Figuro mačka izbriši.



V naslednjih nalogah bomo na platno risali črne in bele pike. Pika bo zato imela dve obleki, črno in belo. Klikni na ikono Pika in izberi zavihek Obleke (Costumes). Klikni na gumb Kopiraj. Nastane nova obleka, ki je enaka prvi. Klikni na gumb Uredi in v Urejevalniku slik s pomočjo orodja Zapolni piko prebarvaj na belo. Obleki poimenuj črna in bela.

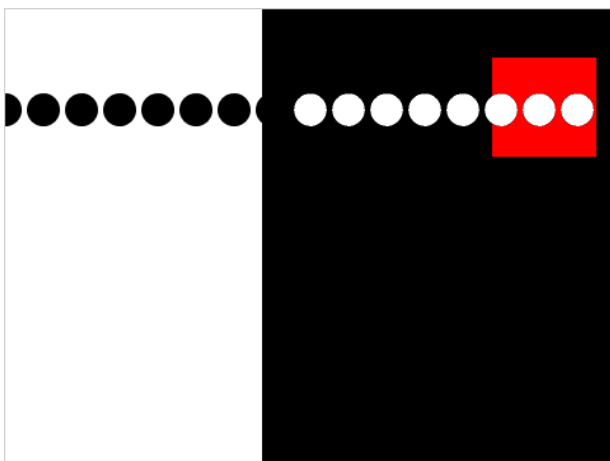


Z ukazom preklopi na videz (switch costume to) med črno in belo piko.

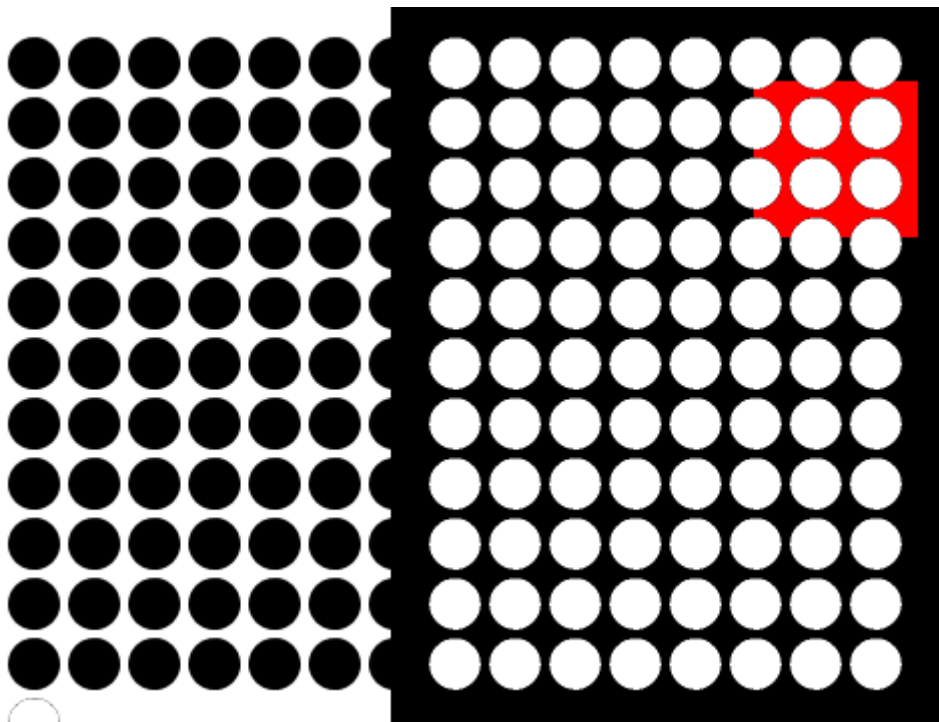


lahko izbiramo

Napiši program, ki nariše celo vrsto pik. Pike naj bodo na belem ozadju črne, na črnem pa bele. Naredi dovolj velik korak, sicer se bo figura dotikala sam sebe in ne bo ustrezno spremenila barve.

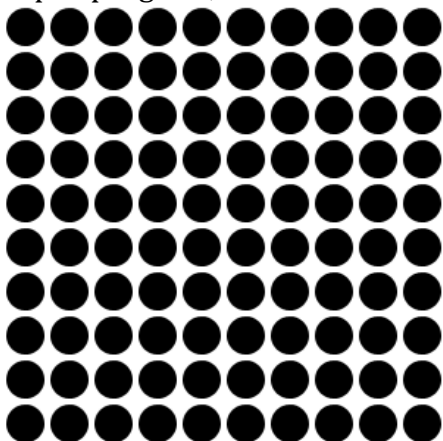


Napiši program tako, da bo ves zaslon prekrit s pikami.



ENA VRSTA RDEČA, DRUGA ČRNA...

Napiši program, ki naredi 10 vrst po 10 črnih pik. Začni v levem spodnjem kotu.



Napiši program, ki bo izmenično narisal eno vrsto črnih in eno vrsto rdečih pik.



Preprosta rešitev bi bila, če bi narisal eno vrsto črnih in eno vrsto rdečih pik in nato vse skupaj petkrat ponovil.

Vendar bomo raje uporabili idejo, ki bo uporabna še marsikdaj. Od spodaj navzgor je vsak liha vrsta črna, vsaka soda pa rdeča. Torej če je vrstica liha bomo uporabili ukaz *Preklopi na videz*

črna , če je vrstica soda, pa ukaz *Preklopi na videz* *rdeča* .

Pri določanju, ali je vrstica soda ali liha, si bomo pomagali s spremenljivko. Če bo vrednost spremenljivke 0, bo vrstica liha, če bo vrednost spremenljivke 1, bo vrstica soda.

Če pogledamo rešitev, vidimo da se v pogojnem stavku vrednost spremenljivke vrstica poveča za ena ali zmanjša za ena in tako skače med 0 in 1.



Med vrednostjo 0 in 1 se lahko premikamo tudi drugače. Poglejmo si primer ostanka pri deljenju z 2:

1: 2 = 0, *ostanek* 1

2: 2 = 1, *ostanek* 0

3: 2 = 1, *ostanek* 1

4: 2 = 2, *ostanek* 0

Itd.

Opazimo, da je ostanek pri deljenju izmenično 1 ali 0. Razmislimo, kako bi to lahko uporabili s pomočjo operatorja ostanek pri deljenju  .

Napiši program, ki bo narisal tako sliko. Uporabimo tri barve: rdeča, črna in rumena.

