

**ACM TEKMOVANJE V
PROGRAMIRANJU Z DELČKI
PIŠEK 2022/23**

**2. šolsko tekmovanje
(delovna verzija)**

Naloge in rešitve

Marec 2023

Izbor, priredba in preoblikovanje tekmovalnih nalog: Programski svet tekmovanja

Razvoj tekmovalnega sistema: ACM Slovenija in UL FMF v sodelovanju s France-IOI

KAZALO NALOG

Tekmovalna kategorija: OŠ 4. – 6. r. ZAČETNIKI

MIŠKA MIMI IN NJEN SIRČEK	1
PES IŠČE KOST	2
DOSTAVA PICE	3
ČEBELICA LETI S CVETA NA CVET	5
PETER NAREDI SNEŽAKA	6
JANKO IN METKA	7

Tekmovalna kategorija: OŠ 4. – 6. r. NAPREDNI

METULJ TEO RIŠE	8
LAČNA ŽABICA	9
GREGOR IŠČE ŠOLO	10
KEKEC V LABIRINTU	11
PREBERI IN POBARVAJ	12
MOTORISTI	13

Tekmovalna kategorija: OŠ 7. – 9. r. ZAČETNIKI

POLŽEK SLINČEK	15
TAČKO IN SKRIVALNICE	16
JANKO IN METKA	18
ČEBELICA MAJA NABIRA CVETNI PRAH	20
ŠKATLA ZA SEMENA	22
SLADOLED	24

Tekmovalna kategorija: OŠ 7. – 9. r. NAPREDNI

ČEBELICA LETI S CVETA NA CVET	26
PREBERI IN POBARVAJ	27
SLADOLED	28
KEKEC V LABIRINTU	30
KOZMONAVT	31
IZBIRA GESLA	32

Tekmovalna kategorija: SŠZAČETNIKI

ZASNEŽENA CESTA	33
ZELENJAVNI VRT	34
PIŠKOV BOŽIČNI OKRAS	35
PIŠKOVE VOLITVE	36
PIŠEK NA ORIENTACIJSKEM TEKMOVANJU	37

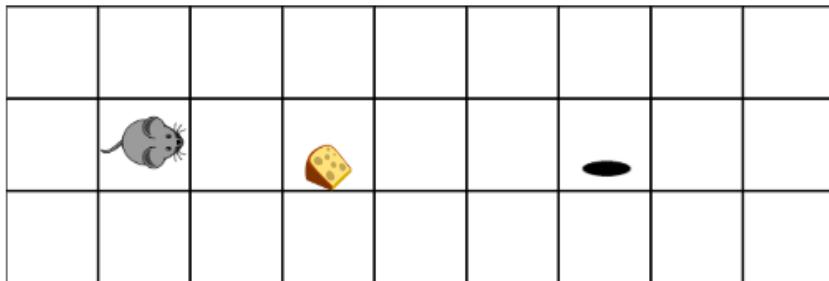
Tekmovalna kategorija: SŠ NAPREDNI

OBEŠANJE POSTELJNINE	38
ZAPIS URE	39
KOZMONAVT	41
NEZADOSTNA ŠTEVILA	42
PIŠEK GLEDA TEKMO	43
Rešitve	45

MIŠKA MIMI IN NJEN SIRČEK

OŠ 4. – 6. r. ZAČETNIKI

Miška Mimi je pred seboj opazila kos sira. Pomagaj ji napisati program tako, da bo prišla do sirčka, ga pobrala, šla v svojo luknjico in ga tam odložila.



Povezava do naloge

Ideja reševanja

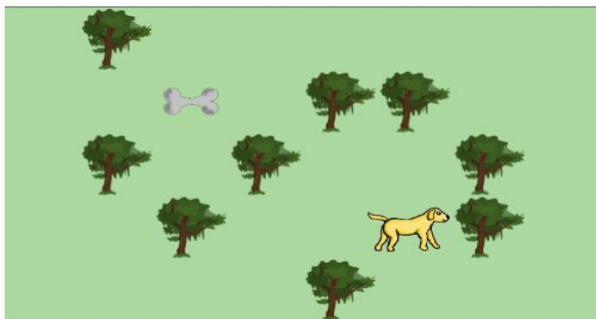
Najprej dobro preberi navodilo in si oglej sliko. Opaziš lahko, da boš za rešitev potreboval tri delčke: premik v desno, poberi sirček in pospravi sirček. Sedaj pa jih le nanizaj v pravilni vrstni red in zaženi program.

Povezava do rešitev

PES IŠČE KOST

OŠ 4. – 6. r. ZAČETNIKI

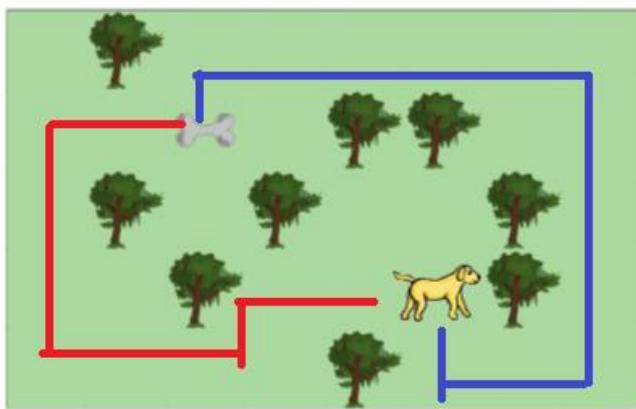
Pes Runo bi rad dobil kost, ki jo je skril med grmičevje. Pomagaj mu jo najti. Runo ne more skozi grme, ampak mora naokoli.



Povezava do naloge

Ideja reševanja

Preberi navodilo in si oglej sliko. Ker so med Runotom in kostjo grmički, moraš napisati program, ki ga bo vodil okrog njih. Spodaj smo označili dve izmed možnih poti, pravilne pa so seveda tudi druge.



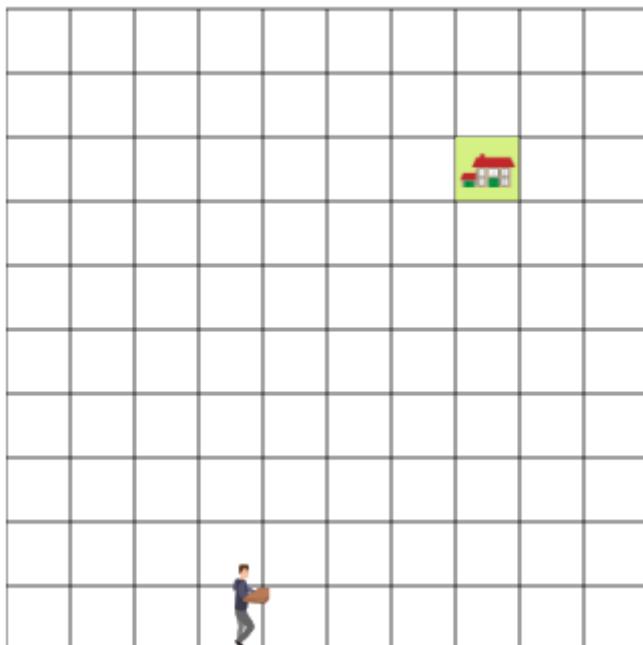
Ne glede katero pot izbereš, program napišeš tako, da delčke nizaš enega za drugim. Za daljše premike v isti smeri lahko uporabiš preprosto zanko in tako program nekoliko skrajšaš. Na koncu ne pozabi dodati delčka, ki kost pobere. Ker slika nima jasno narisane mreže, lahko program med reševanjem večkrat zaženeš in preveriš, ali kuža hodi po pravi poti, in program po potrebi popraviš.

Povezava do rešitev

DOSTAVA PICE

OŠ 4. – 6. r. ZAČETNIKI

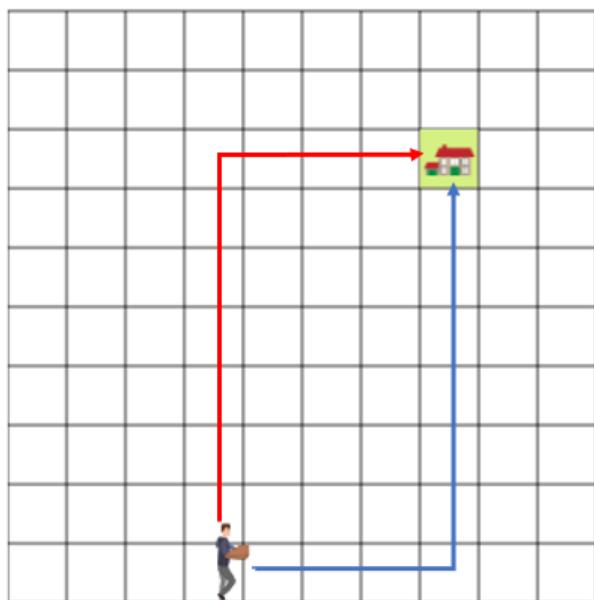
Pica je pečena! Pomagaj dostavljalcu, da jo prinese do hiše.



Povezava do naloge

Ideja reševanja

Edini delček, s katerim premaknemo dostavljalca, je delček za naprej. Zato bomo morali biti vedno obrnjeni v smeri, kamor gremo. Največja težava pri tej nalogi je, da je na voljo le 8 delčkov. Zato bo nujno uporabiti zanke. Z eno se bomo gibali v vodoravni smeri, z drugo pa po navpični. Ali gremo najprej po spodnji vrstici in nato gor do hiše, ali pa gor do vrstice s hišo in potem do nje, je vseeno.



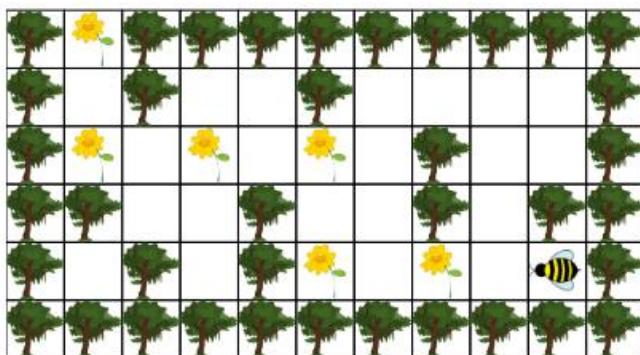
Paziti pa moramo, da ne tavamo preveč, saj nam bo hitro zmanjkalo delčkov!

Povezava do rešitev

ČEBELICA LETI S CVETA NA CVET

OŠ 4. – 6. r. ZAČETNIKI

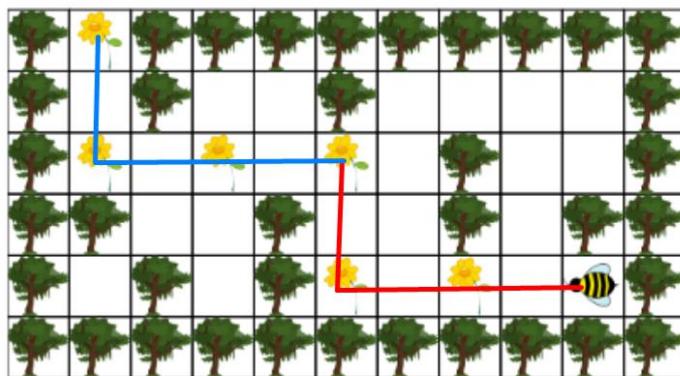
Čebelica leti s cveta na cvet. Napiši čim krajši program, ki bo čebelico popeljal čez ta zaviti travnik.



Ideja reševanja

Čebelico premikamo z nizanjem delčkov. Če imamo več zaporednih enakih delčkov, lahko uporabimo zanko. Pazi le, da ne boš naredil preveč ponovitev, ker se bo drugače čebelica zaletela v drevo!

Nalogo pa lahko rešimo tudi nekoliko drugače. Opazimo namreč lahko vzorec, ki se dvakrat ponovi.



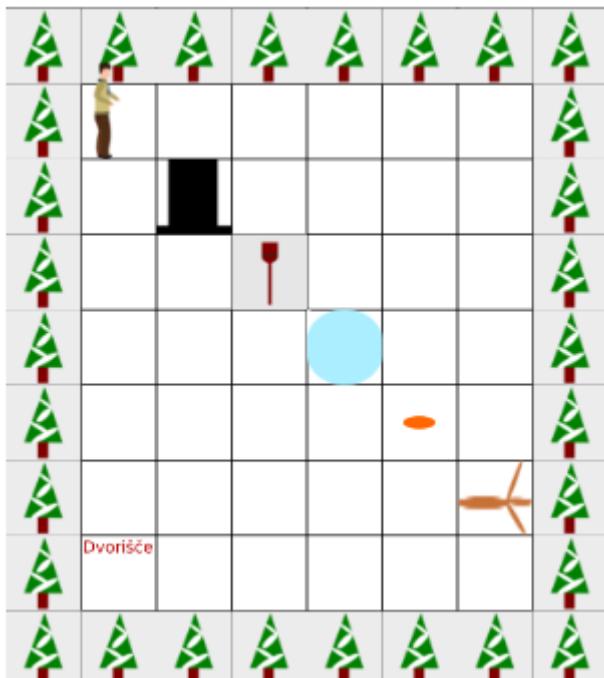
Zato uporabimo zanko, ki se izvede dvakrat, znotraj zanke pa spet s pomočjo zanke naredimo vodoraven in potem navpičen premik.

Povezava do rešitev

PETER NAREDI SNEŽAKA

OŠ 4. – 6. r. ZAČETNIKI

Peter si želi postaviti snežaka pred svojo hišo. Da bi to storil, mora najprej po zasneženem polju poiskati vse njegove sestavne dele. Ko jih pobere, naj snežaka postavi na označenem mestu na dvorišču.



Povezava do naloge

Ideja reševanja

Najprej poberemo vse dele snežaka. Pri tem upoštevamo, da se med vsakim delom premaknemo desno in potem dol. Ko imamo vse delčke, se premaknemo na polje, kjer mora biti snežak in ga postavimo tam.

Povezava do rešitev

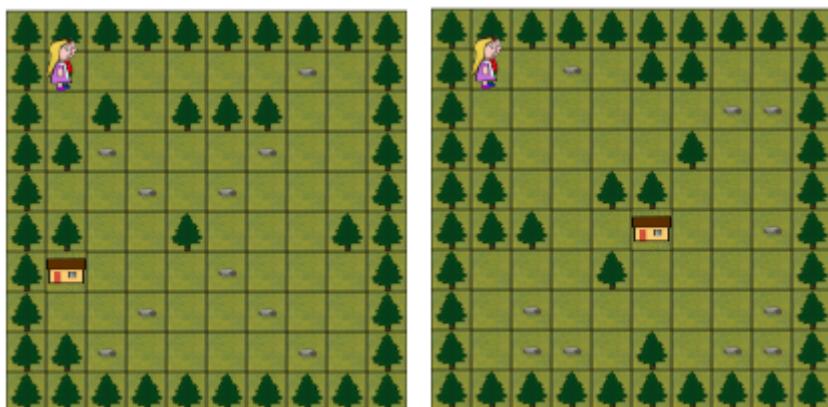
JANKO IN METKA

OŠ 4. – 6. r. ZAČETNIKI

Zagotovo poznaš zgodbo o Janku in Metki. Odpravila sta se v gozd in da se ne bi zgubila, sta si pot do hiše označila z desetimi kamenčki. Zapomnila sta si, da morata vsakič, ko prideta do kamenčka, zaviti desno.

Za vsak primer pa si bosta zapisala podrobna navodila obliku programa. Vsi za program potrebeni delčki so že pripravljeni na delovni površini. Ti jih le sestavi tako, da bosta Janko in Metka prišla nazaj do svoje hiše.

A pazi, isti program mora delovati pravilno tako pri razporeditvi kamenčkov kot na Test 1, kot tudi Test 2.



[Povezava do naloge](#)

Ideja reševanja

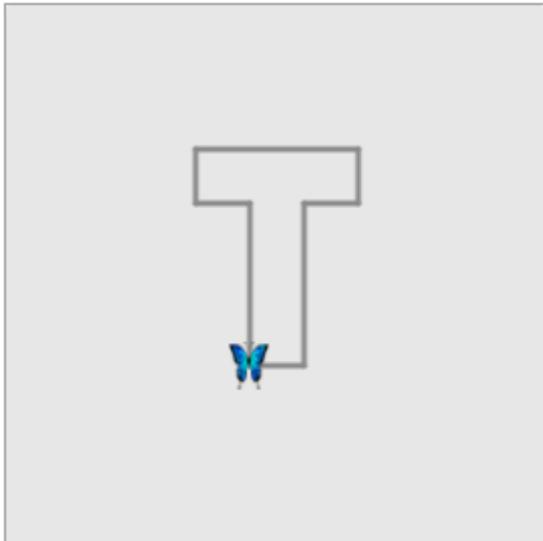
Janko in Metka bosta hodila, dokler ne bosta doma. Zato bomo uporabili zanko "ponavljam dokler ni pri domu". Kot pravijo navodila, se takrat, ko sta pri kamenčku, obrneta desno.

[Povezava do rešitev](#)

METULJ TEO RIŠE

OŠ 4. – 6. r. NAPREDNI

Metulj Teo želi na list papirja narisati črko T. Pomagaj mu. Krajše črte naj bodo dolge 10 enot, daljše pa 30 enot.



[Povezava do naloge](#)

Ideja reševanja

Ker Teo že gleda navzgor, bomo začeli kar z navpično črto. Zasukali se bomo v levo, narisali črto dolžine 10, se zasukali desno in narisali še en kratko črto. Ko se zasukamo v desno, pride na vrsto daljša črta. Potem dvakrat ponovimo zasuk v desno in risanje kratke črte. Po zasuku v levo narišemo dolgo črto. Črko T dokončamo z zasukom v desno in premikom naprej za 10.

[Povezava do rešitev](#)

LAČNA ŽABICA

OŠ 4. – 6. r. NAPREDNI

Žabica Rega je lačna. Pomagaj ji, da poje vse muhe. Skače lahko le po lokvanjih, na koncu naj se vrne na začetno pozicijo.



Povezava do naloge

Ideja reševanja

Vidimo, da so lokvanji razporejeni kot stranice kvadrata. Zato bomo Rego s pomočjo zanke, ki se ponovi štirikrat, poslali na potovanje preko vseh 4 stranic. Na vsaki stranici skoči trikrat in če naleti na muho, jo z užitkom poje.

Povezava do rešitev

GREGOR IŠČE ŠOLO

OŠ 4. – 6. r. NAPREDNI

Gregor se bo udeležil tekmovanja iz programiranja na šoli v Piškoviji. A ker bo na tej šoli prvič, bo sledil navodilu, ki mu ga je podal Mitja:

- Pojdi naprej, dokler ne prideš do reke.
- Nadaljuj pot po stezi. Sledi puščicam, dokler ne prideš do smreke.
- Obrni se levo in nadaljuj pot do šole.

A ker je Gregor pravi programer, bi rajši imel navodila v obliki programa. Zato napiši program in tako pomagaj Gregorju najti šolo.



Povezava do naloge

Ideja reševanja

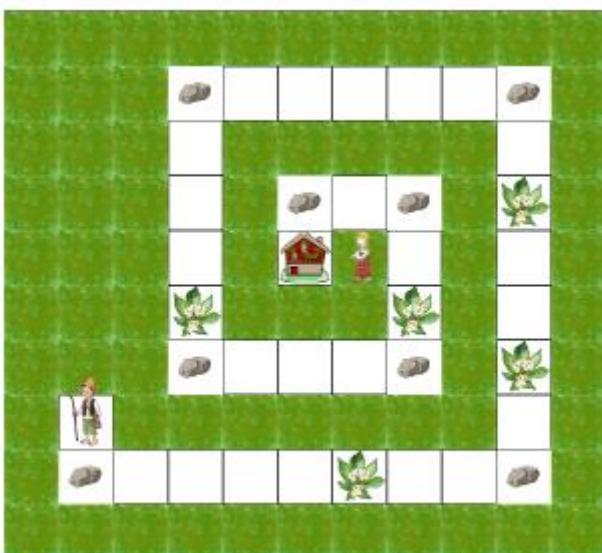
Program bo najlažje sestaviti tako, da bomo sledili navodilom. Z zanko *ponavljam dokler ni* gremo do reke. Nato uporabimo isto zanko, a ukaz premikaj se v smeri puščic, da pridemo do smreke. Obrnemo se proti šoli in se premikamo naprej, dokler ne pridemo v šolo.

Povezava do rešitev

KEKEC V LABIRINTU

OŠ 4. – 6. r. NAPREDNI

Pehta je zaprla Mojco v svojo kočo, skrito globoko v labirintu. Pomagaj Kekcu najti pot do Pehtine koče. A kako? Na srečo mu je Kosobrin povedal, da pot zavije vedno, ko na poti naleti na kamen. Napiši program, ki bo Kekca pripeljal do koče. Na poti mora Kekec pobrati še vse zdravilne rože, ki jih najde.



Povezava do naloge

Ideja reševanja

Nalogo lahko rešimo na dva različna načina. Pri prvem uporabimo le eno zanko. Premikamo se, dokler ne pridemo do Pehtine koče, ter sproti pobiramo rože ter se po potrebi (ko smo na kamnu) zasukamo v levo.

Drugi način uporablja dvojno zanko. Zunanja zanka se izvaja, dokler ne pridemo do Pehte, ali pa upoštevamo to, da je pot sestavljena iz osmih delov. V notranji zanki se premikamo, dokler ne pridemo do kamna ter ob tem pobiramo rože. Ko smo na kamnu pa se zasukamo in vstopimo v naslednji del.

Povezava do rešitev

PREBERI IN POBARVAJ

OŠ 4. – 6. r. NAPREDNI

Zmajček barva določene vrstice v rumenem pravokotniku. V zeleno barvo sme pobarvati le tiste vrstice pravokotnika, pred katerimi je napisano sodo število. Sestavi program, ki bo usmerjal zmajčka pri barvanju.

							
5							
2							
3							
8							
7							
4							

							
22							
200							
25							
91							
105							
78							

							
201							
506							
88							
53							
302							
510							

Povezava do naloge

Ideja reševanja

Najprej dobro preberi navodila in si oglej vse testne primere. Skupno jim je, da mora zmajček vedno prebrati natanko šest števil, ki so v prvem stolpcu. To pomeni, da lahko uporabimo zanko, ki se bo izvedla 6-krat. Vanjo pa bomo vstavili en delček, ki bo zmajčka premaknil za polje navzdol, nato pa pogojni stavek. Preverjali bomo pogoj, ali je številka na polju soda - za preverjanje tega uporabimo delčke, ki so na voljo v razdelku "Matematika". In če ta pogoj velja, se bo zmajček 7-krat premaknil v desno in ob tem še pobarval polje. Nato pa se mora še vrniti 7 polj nazaj v levo, da bo lahko nadaljeval z novo ponovitvijo zanke v začetnem prvem stolpcu. Če število na polju ni sodo, zmajček ne bo ničesar barval, ampak bo skočil v novo ponovitev zanke - torej se bo premaknil dodatno polje navzdol in znova preverjal sodost oz. lihost števila na polju.

Povezava do rešitev

MOTORISTI

OŠ 4. – 6. r. NAPREDNI

Pišek je navdušen nad motorji, zato steče proti motorjem. Žal se spotakne in z motorjev popadajo čelade, ki na srečo pristanejo tik pred motorji. Pomagaj mu postaviti čelade nazaj na motorje. Seveda se morajo barve ujemati. Pišek lahko nosi le eno čelado naenkrat.



Povezava do naloge

Ideja reševanja

Najprej dobro preberi navodila in si oglej vse tri testne primere. Opaziš lahko nekaj podobnosti, in sicer se čelada vedno nahaja eno polje levo od motorja, in pa vsi trije testni primeri imajo enako veliko mrežo, ki jo mora Pišek prehoditi.

Piška premaknemo eno polje navzgor (prve vrstice ni potrebno pregledovati, ker je prazna v vseh treh testnih primerih). Nato pa ga 5-krat premaknemo v levo, pri vsakem premiku s pomočjo pogojnega stavka preverimo, če je Pišek na čeladi. Če je na čeladi, jo pobere, se premakne za polje v desno, jo odloži na motor in se premakne nazaj za polje v levo. Nato nadaljuje svojo pot po vrstici. Ko Pišek pride do konca vrstice, se z zanko ponovno vrne 5 polj v desno, na začetek. Ker bo Pišek za vsako izmed vrstic naredil enak postopek, lahko uporabimo gnezdeno zanko, ki jo bomo izvedli 5-krat. Z zunanjim zankom Piška premikamo po vrsticah počasi navzgor, z notranjimi zankami pa iščemo čelade in jih premikamo na motorje.

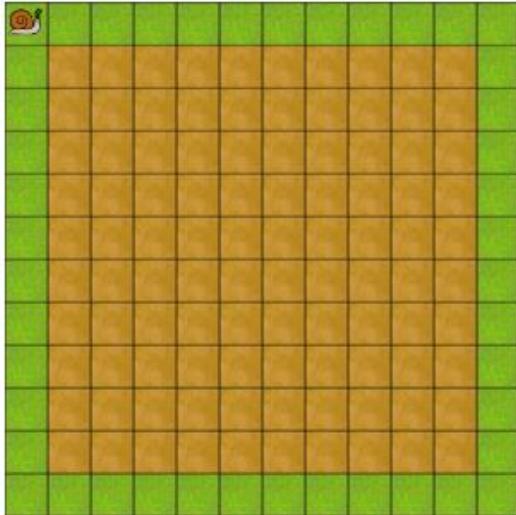
Nalogo lahko rešimo tudi nekoliko drugače - torej da se Pišek premika po stolpcih navzgor ali pa se premika od desne proti levi. Možnih je torej več rešitev, pri vseh pa je ideja reševanja precej podobna.

Povezava do rešitev

POLŽEK SLINČEK

OŠ 7. – 9. r. ZAČETNIKI

Polži pri premikanju puščajo sluz zato, da se lažje gibljejo, pa tudi da označijo svoje območje. Polžek Slinček bi rad obkrožil svoj vrt in ga označil. Pomagaj mu in sestavi navodila, kako naj se giblje.



Povezava do naloge

Ideja reševanja

Najprej si dobro oglej sliko. Napisati moraš program, ki bo polžka vodil okrog vrta, torej po stranicah kvadrata. Začetno smer premikanja si lahko izbereš sam:
1) polžek začne svojo pot najprej v desno in se premika v smeri urinega kazalca
2) polžek začne svojo pot navzdol in se premika v obratni smeri urinega kazalca.
Obe rešitvi sta pravilni.

Ker je kvadrat precej velik in bi za nizanje delčkov za premikanje porabil precej časa, raje uporabi zanko. Preštej število premikov, ki jih mora polžek opraviti, da se premakne skozi eno stranico kvadrata. To število nato zapiši v število ponovitev v zanki. Zaporedno uporabi 4 zanke - po eno za vsako stranico.

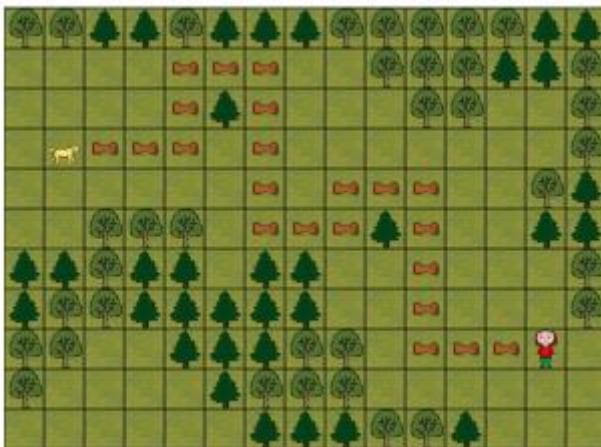
Na koncu moraš še poskrbeti, da polžek na vsakem polju pusti sled. Za to moraš uporabiti poseben delček. Sled lahko polžek pusti pred premikom ali pa po njem. Obe rešitvi sta pravilni.

Povezava do rešitev

TAČKO IN SKRIVALNICE

OŠ 7. – 9. r. ZAČETNIKI

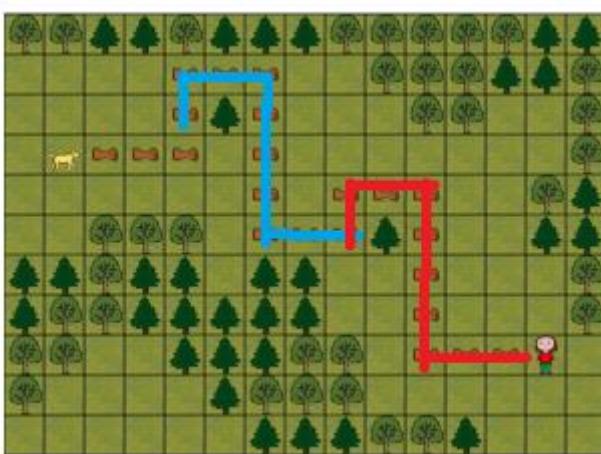
Marko se v gozdu s svojim psičkom Tačkom rad igra skrivalnice. Napiši program, ki bo Tačkov izostreni smrček vodil do vseh priboljškov in ga pripeljal do svojega lastnika. Število delčkov, ki jih lahko uporabiš, je omejeno, zato si dobro ogled sliko in poišči vzorec Tačkovega gibanja.



Povezava do naloge

Ideja reševanja

Najprej si dobro ogled sliko. Ker moraš napisati čim krajši program, poskusni na Tačkovi poti poiskati vzorec, ki se ponavlja. Vzorec je označen spodaj - prva ponovitev z modro barvo in druga ponovitev z rdečo.



Tačka moramo najprej premakniti do novega začetnega polja torej tri polja v desno in eno polje navzgor. Potem pa lahko zapišemo del kode, ki se bo dvakrat

ponovil. Za dvojne premike v desno lahko uporabimo zanko, prav tako za 4 premike navzdol. Ker se rdeči del poti ponovi, lahko modri del poti, ki smo ga ravno kar zapisali, vstavimo v zanko, ki se ponovi dvakrat. Na koncu moramo dodati še en premik v desno izven zanke, da bo Tačko prišel točno na polje, na katerem stoji Marko.

Nalogo lahko rešimo tudi brez dvojne zanke, torej z nizanjem delčkov premikanja v pravo zaporedje in uporabo zanke za 4 premike navzdol.

Povezava do rešitev

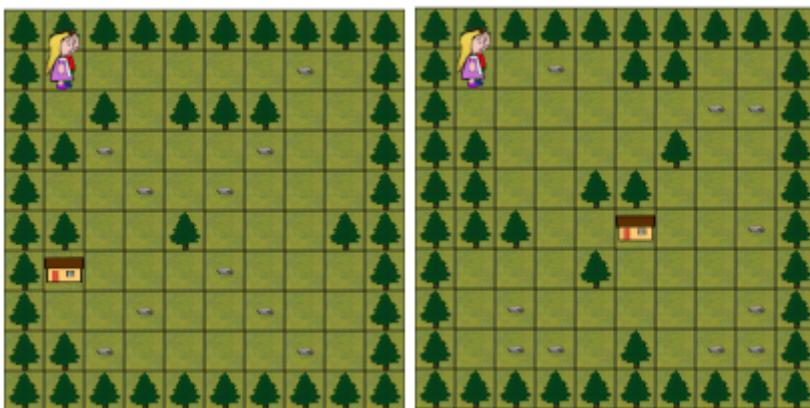
JANKO IN METKA

OŠ 7. – 9. r. ZAČETNIKI

Zagotovo poznaš zgodbo o Janku in Metki. Odpravila sta se v gozd in da se ne bi zgubila, sta si pot do hiše označila z desetimi kamenčki. Zapomnila sta si, da morata vsakič, ko prideta do kamenčka, zaviti desno.

Za vsak primer pa si bosta zapisala podrobna navodila obliki programa. Vsi za program potrebeni delčki so že pripravljeni na delovni površini. Ti jih le sestavi tako, da bosta Janko in Metka prišla nazaj do svoje hiše.

A pazi, isti program mora delovati pravilno tako pri razporeditvi kamenčkov kot na Test 1, kot tudi Test 2.



Povezava do naloge

Ideja reševanja

Najprej si dobro oglej obe slike. Opaziš lahko, da sta si precej različni, torej moraš napisati univerzalni program, ki bo Janka in Metko v obeh primerih pripeljal do hiše.

Še enkrat podrobno preberi navodilo. Ugotoviš lahko dve stvari:

- 1) vsi delčki, ki jih potrebuješ, so že na delovni površini, zato jih ne umikaj, ker si nalogo po nepotrebnem otežiš z iskanjem delčkom v izbirnem polju na levi;
- 2) v navodilih imaš zapisano pravilo, kaj morata Janko in Metka storiti, da prideta do doma - torej vsakič, ko prideta do kamenčka, morata zaviti desno. Poskusimo to pravilo sedaj zapisati v obliki programa.

Na delovni površini že imamo delček "ponavljam dokler". Janko in Metka se premikata naprej, dokler nista pri domu. Delček "pri domu" torej vstavimo v pogoj zanke, premik naprej pa v zanko. To pomeni, da se bo zanka izvajala vse dokler ne bosta dosegla polja s hišo.

Ostane pa nam še nekaj delčkov. Še enkrat preberimo pravilo - ko prideta do kamenčka, zavijta desno. To moramo pretvoriti v programsko kodo. Uporabimo torej pogojni stavek, če je na polju kamenček, izvedemo obrat v desno.

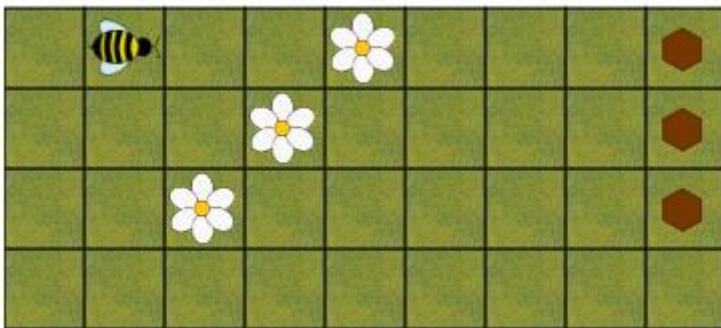
Na koncu moramo pogojni stavek vstaviti v našo zanko. Ker Janko in Metka v nobenem testnem primeru ne začenjata poti na polju s kamenčkom, lahko pogojni stavek v zanko vstavimo pred ali pa za premikom naprej. V obeh primerih program deluje pravilno.

Povezava do rešitev

ČEBELICA MAJA NABIRATI CVETNI PRAH OŠ 7. – 9. r. ZAČETNIKI

Čebelica Maja se uči nabirati cvetni prah. A ker je še začetnica, potrebuje natančna navodila. Za Majo napiši navodila v obliki programa ter pri tem upoštevaj sledeče:

- Maja lahko naenkrat nosi le cvetni prah nabran na enem cvetu.
- Maja lahko s cvetnim prahom ene cvetlice napolni le en sat.



Povezava do naloge

Ideja reševanja

Nalogo lahko rešimo na veliko različnih načinov. Programersko lep način je z uporabo dvojne zanke.

1) Najprej napišimo kodo za premikanje po prvi vrstici. Uporabimo zanko “dokler ni” in za pogoj najprej nastavimo “na cvetu”, v zanko pa vstavimo premikanje v desno. V tej zanki se bo čebelica premikala desno toliko časa, dokler ne bo prišla do prve cvetlice. Ko bo na cvetu, pa naj pobere cvetni prah. Drugo zanko naredimo podobno, le da se izvaja, dokler čebelica ni na panju, prav tako se premika v desno, na koncu pa tokrat odloži cvetni prah. Čebelico z enostavno zanko na koncu še premaknimo nazaj v levo na začetno mesto. Nato pa izven zanke čebelico premaknimo v spodnjo vrstico.

2) Ker smo napisali univerzalno kodo, po kateri bo čebelica pravilno pobrala in odložila cvetni prah ne glede na to, kako sta si oddaljeni cvetlica in sat, jo lahko zdaj le izvedemo trikrat - za vsako vrstico posebej. Uporabimo še eno zanko, v katero vstavimo kodo, ki smo jo ravnomerno zapisali za reševanje ene vrstice.

Lahko pa uporabimo tudi bolj preprosti način z nizanjem delčkov v zaporedje in uporabo enojnih zank. Najprej čebelico premaknemo do cvetlice z uporabo zanke in premikom v desno. Na cvetlici čebelica pobere cvetni prah, nato pa zapišemo novo zanko s premiki v desno in na panju čebelica odloži cvetni prah. Z zanko premaknemo čebelico znova nazaj v levo in podobno storimo še v naslednjih dveh vrsticah. Ta rešitev je sicer možna, a zahteva kar nekaj časa in veliko nizanja delčkov.

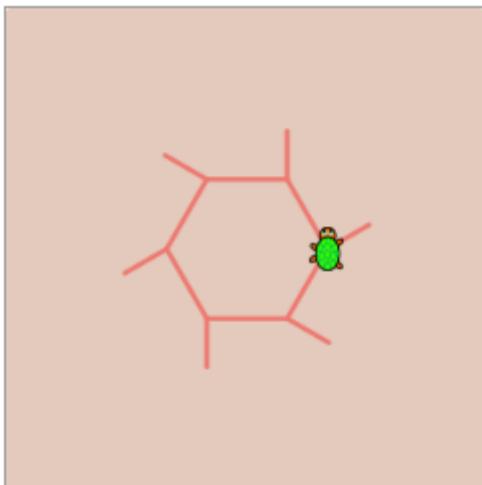
Povezava do rešitev

ŠKATLA ZA SEMENA

OŠ 7. – 9. r. ZAČETNIKI

Želva bi rada iz lepenke izdelala posodo za semena. Ta naj ima za osnovo šestkotnik. Uspelo ji je skicirati plašč škatle (glej sliko), sedaj pa mora označiti robove, kjer bo lepenka upognjena. Ti naj bodo živo rdeči.

Dolžina stranice šestkotnika je 100 enot, škatla pa bo visoka 60 enot. Vendar pozor! Bodimo varčni tudi z barvo in ne pobarvajmo dvakrat isti rob (namig: uporabi delček dvigni/spusti pisalo).



Povezava do naloge

Ideja reševanja

Dobro si oglejmo sliko in preberimo navodilo. Vidimo lahko šestkotnik, z dolžino stranice 100 enot. Poleg šestkotnika pa so tudi kraki škatle, ki merijo 60 enot. Sedaj, ko vemo, kolikšne dolžine so posamezne črte, ne rabimo izgubljati časa s preizkušanjem delovanja programa. Šestkotnik nas tudi spomni, da bomo najverjetneje lahko uporabili zanko, ki se bo ponovila 6-krat. Torej moramo le razmisliti, kako bomo premikali želvo po eni stranici in kraku šestkotnika, to pa bomo nato šestkrat ponovili.

Najprej nastavimo barvo pisala na rdečo in želvo obrnimo v začetni položaj v smer risanja (rišemo lahko v smeri urinega kazalca, ali pa v obratni smeri), npr. v obratno smer urinega kazalca (torej za 30° v levo).

Želva mora najprej spustiti pisalo, da bo puščala za sabo sled. Premaknemo jo po prvi stranici, torej 100 enot naprej. Nato pa zapišemo program, da bo želva narisala prvi krak škatle, obrniti se mora za 30° v desno (to lahko ugotoviš tudi s poskušanjem) in se premakniti 60 enot naprej. Ker želva varčuje z barvo, po isti črti ne sme risati 2-krat, zato dvignemo pisalo in se premaknemo nazaj za 60 enot. Želvo moramo obrniti v smer naslednje stranice šestkotnika, torej 90° v levo. Enak postopek ponovimo za vsako stranico, zato lahko uporabimo zanko, ki jo izvajamo šestkrat.

Možne so tudi nekoliko drugačne rešitve, odvisno kaj vzamemo za osnovo, ki se ponavlja v zanki. Zaporedje ukazov v zanki se lahko razlikuje predvsem v tem, kdaj se odločimo spuščati pisalo (na začetku ali koncu zanke) in pa kdaj se odločimo obračati v smer risanja stranice šestkotnika.

Povezava do rešitev

SLADOLED

OŠ 7. – 9. r. ZAČETNIKI

Pišek je zgradil avtomat za prodajo sladoleda. Sedaj bi rad imel program, ki bo kupcem sladoleda, z odgovorom »DA/NE«, povedal, če je določen okus sladoleda na voljo. Pri tem upoštevaj, da Pišek prodaja samo štiri okuse in sicer »jagoda«, »čokolada«, »borovnica« in »pistacija« Primer izpisa:

Vhod:
jagoda
jagoda
vanilija
čokolada

Izhod:
DA
DA
NE
DA

Pazi, da tudi na koncu zadnjega odgovora skočiš v novo vrstico!

Da bo lažje, je osnovna shema programa že napisana.

Vhod:
jagoda
jagoda
vanilija
čokolada

Izhod:

Vhod:
vanilija
čokolada

Izhod:

Vhod:
vanilija
vanilija
kokos

Izhod:

Vhod:
jagoda
pistacija

Izhod:

[Povezava do naloge](#)

Ideja reševanja

Najprej si dobro preberi navodila naloge in si oglej primer pravilne rešitve. Napisati moraš torej program, ki bo prebral vrstico za vrstico, preveril, ali je zapisan okus sladoleda na voljo, in zapisal odgovor DA/NE - vsakega v svojo vrstico. Oglej si še testne primere. Opaziš lahko, da število besed ni vedno enako. Ta problem nam pomaga rešiti že začetni nastavek programske kode, ki je podana vnaprej - in sicer zanka "dokler ni konec vhoda".

Preostanek problema razdelimo na manjše kose.

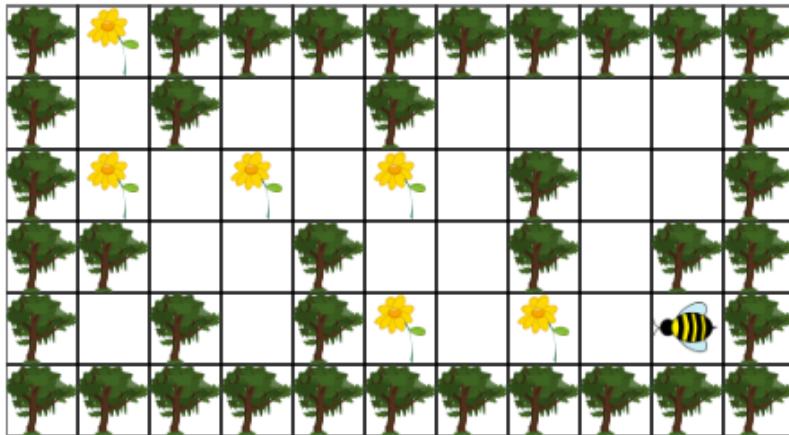
- 1) Da si bo program zapomnil, katero besedo je ravnokar prebral, moramo uporabiti spremenljivko. Med delčki, ki so na voljo na levi strani, imamo že poimenovano spremenljivko "vrstica", lahko pa si ustvarimo tudi svojo. Vrednost spremenljivke nastavimo na prebrano vrstico. Ker bomo njeno vrednost v vsaki vrstici spremenili, jo vstavimo v zanko, ki teče po vrsticah.
- 2) V drugem koraku bomo nato preverili, ali je vrednost, ki smo jo shranili v spremenljivko, enaka enemu izmed štirih sladoledov, ki jih ima Pišek na voljo. Uporabili bomo torej pogojni stavek in delčke, ki so na voljo v razdelku "Logika" in preverjajo enakost. Ker imamo na voljo štiri sladolede, moramo za vsak okus uporabiti svoj delček za preverjanje enakosti, ki pa jih med seboj povežemo z "ali", saj so na voljo vsi štirje okusi. Na koncu torej izpišemo "DA", če je vrednost spremenljivke enaka enemu izmed štirih okusov sladoledov, oz. "NE" v nasprotnem primeru.
- 3) Na koncu pogojnega stavka vstavimo še skok v novo vrstico, ker je vsak zapis na izhodu, v svoji vrstici.

Povezava do rešitev

ČEBELICA LETI S CVETA NA CVET

OŠ 7. – 9. r. NAPREDNI

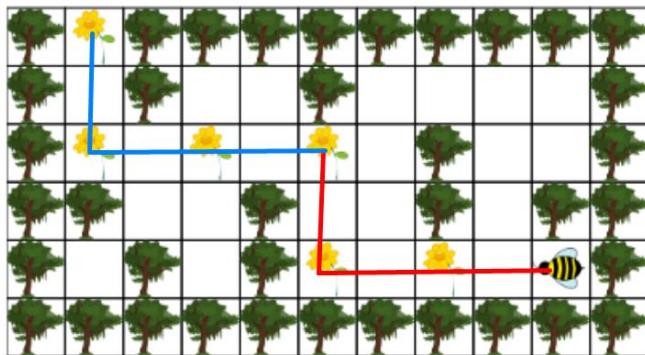
Čebelica leti s cveta na cvet. Napiši program, ki bo čebelico po najkrajši poti popeljal čez travnik, tako da bo obiskala vse cvetlice.



Povezava do naloge

Ideja reševanja

Če poskusimo kar premikati čebelico in pri več zaporednih premikih naprej uporabimo zanko, bomo porabili preveč delčkov. Lahko pa opazimo, da čebelica dvakrat ponovi enak vzorec.



Zato uporabimo zanko, ki se izvede dvakrat, znotraj zanke pa spet s pomočjo zanke naredimo vodoraven in potem navpičen premik. Tako bo čebelica obiskala vse cvetlice, navodila za njeno premikanje pa ne bodo porabile preveč delčkov.

Povezava do rešitev

PREBERI IN POBARVAJ

OŠ 7. – 9. r. NAPREDNI

Zmajček barva vrstice, ki imajo v prvem polju zapisano število. Zmajček prebere število in preveri ali je število sodo ali liho. Če je število sodo, pobarva vsa polja brez številke v tej vrstici z zeleno barvo, če pa je število liho, jih pobarva z rdečo barvo. Sestavi program, ki bo pravilno pobarval vrstice.

5						
2						
3						
8						
7						
4						

22						
200						
25						
91						
105						
78						

201						
506						
88						
53						
302						
510						

Povezava do naloge

Ideja reševanja

Najprej dobro preberi navodila in si oglej vse testne primere. Skupno jim je, da mora zmajček vedno prebrati natanko šest števil, ki so v prvem stolpcu. To pomeni, da lahko uporabimo zanko, ki se bo izvedla 6-krat. Vanjo pa bomo vstavili en delček, ki bo zmajčka premaknil za polje navzdol, nato pa pogojni stavek. Preverjali bomo pogoj, ali je številka na polju soda, tem primeru bomo vrstico pobarvali zeleno, sicer pa rdeče. V obeh primerih moramo zmajčka nato vrniti 7 polj nazaj v levo, da bo lahko nadaljeval z novo ponovitvijo zanke v začetnem prvem stolpcu.

Povezava do rešitev

SLADOLED

OŠ 7. – 9. r. NAPREDNI

Pišek prodaja sladoled. Pomagaj mu in sestavi program, ki bo izračunal ceno izbranega sladoleda. Pri tem upoštevaj da:

- ena kepica stane 1,2 €
- ena kepica z dvema okusoma (pol pol) stane 1,5 €
- sladki kornet stane 0,3 €

Vhod: kepica pol pol sladki kornet kepica	Izhod: 4.2		
Vhod: kepica pol pol sladki kornet kepica	Izhod:	Vhod: kepica pol pol	Izhod:
Vhod: pol pol pol pol sladki kornet	Izhod:	Vhod: kepica kepica	Izhod:

Povezava do naloge

Ideja reševanja

Najprej dobro preberi navodila in si oglej testne primere. Predstavljaš si lahko, da je Pišek natakar, ki prevzame naročilo sladoleda pri vsaki mizi. Na listek si zapisuje, katere article so stranke naročile. Da ne piše celih povedi, zapiše samo kratke oporne točke: pol pol, sladki kornet in pa kepica. Tvoja naloga je torej, da napišeš program, ki bo te besede prebral in jih povezal s pravo ceno. Na koncu

pa mora program seštetи cene vseh artiklov na seznamu, da bo Pišek lahko izstavil račun.

Lotimo se pisanja programske kode z delčki. Ker bomo seštevali cene posameznih artiklov, moramo uporabiti spremenljivko (npr. "znesek"). Določimo si jo na začetku, njeno vrednost pa nastavimo na 0. Po prebrani vsaki vrstici bomo vrednost te spremenljivke povečali za ustrezno vsoto. Uporabimo lahko delček "spremeni za", ali pa običajne delčke seštevanja. Obe možnosti sta prikazani med rešitvami v drugem delu brošure.

Sedaj pa se moramo le še premakniti po vrsticah in vsaki besedi določiti pravilno ceno. Določiti moramo drugo spremenljivko (npr. "vrstica"), v katero shranimo vrednost, ki jo je program prebral. S pogojnimi stavki lahko preverjamo ali je beseda, ki smo jo shranili v spremenljivki "vrstica" enaka kateremu od naših danih artiklov. Preverjamo lahko vse 3 pogoje, ali pa le 2 in namesto tretjega uporabimo "sicer", saj opazimo, da v nobenem testnem primeru nimamo neznanih artiklov, ki jim ne znamo določiti cene.

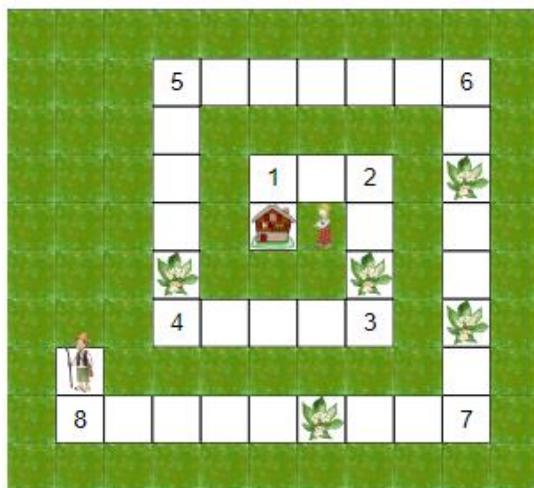
Ker imamo različno število vrstic v posameznih testnih primerih, moramo uporabiti zanko, ki se izvaja vse dokler ni konec vhoda. V zanko torej vstavimo pogojne stavke in ustrezno spremiščanje naše spremenljivke "znesek". Na koncu ne pozabimo še izpisati končne vrednosti našega "zneska".

Povezava do rešitev

KEKEC V LABIRINTU

OŠ 7. – 9. r. NAPREDNI

Pomagaj Kekcu najti pot do Pehtine koče. Pehtina koča je skrita globoko v labirintu. Na vsakem ovinku je polje, ki je označeno s številko, ki ti pove dolžino stranice, ki jo mora Kekec prehoditi. Sestavi program, ki bo Kekca pripeljal do koče in po poti pobral še vse zdravilne rože.



Povezava do naloge

Ideja reševanja

Najprej dobro preberi navodila in si oglej sliko. Opaziš lahko, da boš moral program napisati tako, da bo Kekec prebral število v polju, se tolikokrat premaknil naprej, vmes pa še pobral rožo, če bo naletel nanjo na poti.

Kekca torej premaknemo naprej do prvega polja s številom. Preberemo število in ga shranimo v spremenljivko (npr. "stevilo"). Kekca obremo v levo stran, nato pa uporabimo zanko, ki se izvede tolikokrat, kolikor je vrednost "stevila". V zanko vstavimo še pogojni stavek, s katerim preverimo, ali je na vsakem polju roža, in če je, jo pobremo. Ker vidimo, da imamo na sliki 8 števil, torej našo zanko vstavimo v zunanjo zanko, ki jo izvedemo 8-krat.

Druga možnost reševanja je, da ne prebiramo vsakega števila posebej, saj opazimo, da je vsako naslednje število natanko za 1 manjše od prejšnjega.

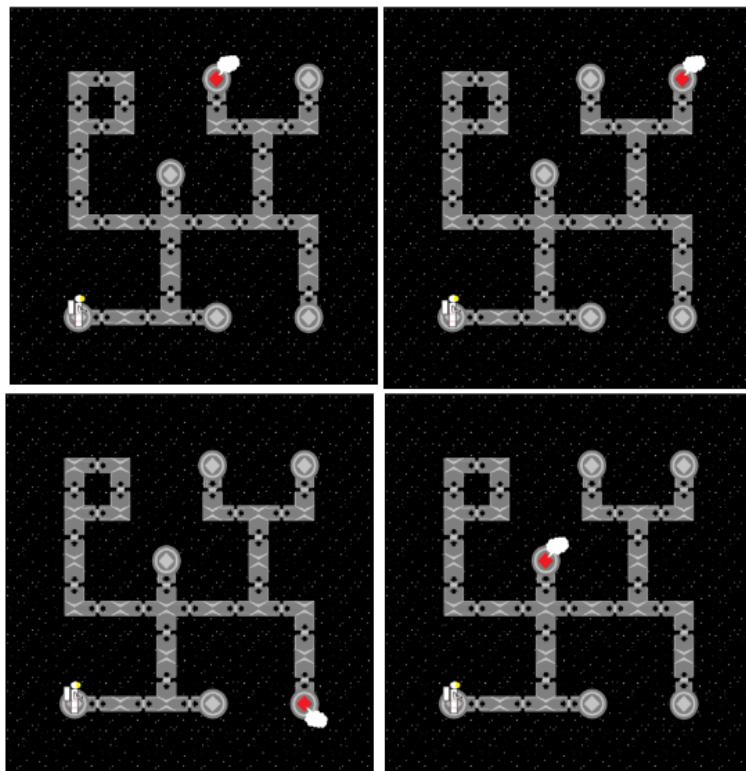
Povezava do rešitev

KOZMONAVT

OŠ 7. – 9. r. NAPREDNI

Kozmonavt živi na vesoljski postaji. Iz ene strani postaje na drugo, se giblje po ceveh, kjer odpravlja okvare. Danes so se pojavile štiri okvare (poglej različne teste). Pomagaj kozmonavtu priti do okvar in jih odpraviti.

Namig: pri iskanju poti, se najprej obrni v levo, nato naravnost in še v desno.



Povezava do naloge

Ideja reševanja

Najprej dobro preberemo navodila. Naloga morda na prvi pogled deluje zelo kompleksna, a če sledimo navodilom, bo kozmonavt vedno našel okvaro. Program moramo izvajati dokler ne kozmonavt ne pride do polja z okvaro, zato uporabimo zanko. Vanjo pa vstavimo pogojne stavke. Prva pot, ki jo kozmonavt vedno izbere je pot v levo, nato sledi pot naravnost, in na koncu v desno. Če ne more nikamor, se vedno lahko vrne nazaj (torej se 2-krat obrne v desno ali levo). Tem pravilom sledi na vsakem koraku.

Povezava do rešitev

IZBIRA GESLA

OŠ 7. – 9. r. NAPREDNI

Pišek si želi izmisliti močno geslo, s katerim želi zavarovati svoj profil na družbenem omrežju Pero, ki je eno najbolj priljubljenih med njegovimi pernatimi prijatelji. Ker je slišal, da je dolga gesla težko uganiti, se je odločil, da bo za geslo izbral kar cel stavek.

Napiši program, ki bo prebral stavek, obrnil vrstni red besed in odstranil vse presledke. V programu uporabi že napisano funkcijo obrni.

Vendar pazi! Nekateri delčki v funkciji manjkajo. Poišči jih in jih postavi na pravo mesto tako, da bo funkcija pravilno delovala.

Vhod: Pišek je najboljši	Izhod:
Vhod: Vsi radi tekmujejo	Izhod:

Povezava do naloge

Ideja reševanja

Najprej dobro preberemo navodila. Na delovni površini imamo že napisan program, ki pa mu manjka nekaj delčkov. Opazimo lahko, da nam manjkata dva dela v obeh pogojnih stavkih. V obeh primerih moramo vrednost spremenljivke "besedilo" spremeniti za isto vrednost - in sicer bomo besedo zapisali pred trenutno besedilo. Na koncu funkcije pa ne smemo pozabiti izpisati zadnje vrednosti, ki smo jo shranili v spremenljivki "besedilo".

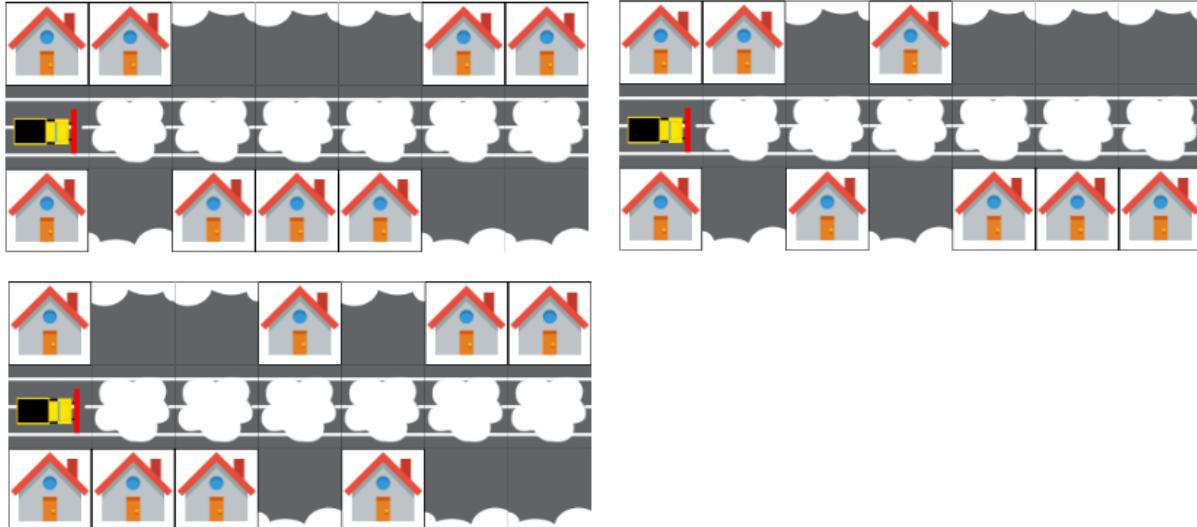
Ko je funkcija popravljena, pa nam manjka še najpomembnejše. Poklicati jo moramo v programu. In sicer bomo izpisali njeno vrednost, ko preberemo vrstico.

Povezava do rešitev

ZASNEŽENA CESTA

SŠ ZAČETNIKI

Ulico je pobelil sneg. Voznik pluga naj odloži sneg na tisto stran ulice, kjer ni hiše.



Povezava do naloge

Ideja reševanja

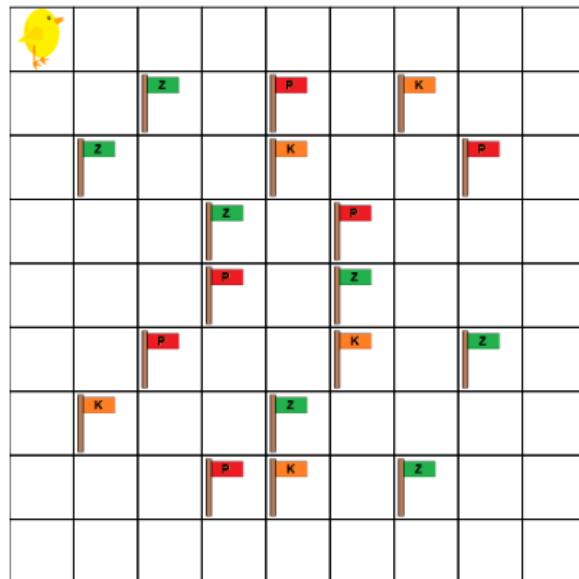
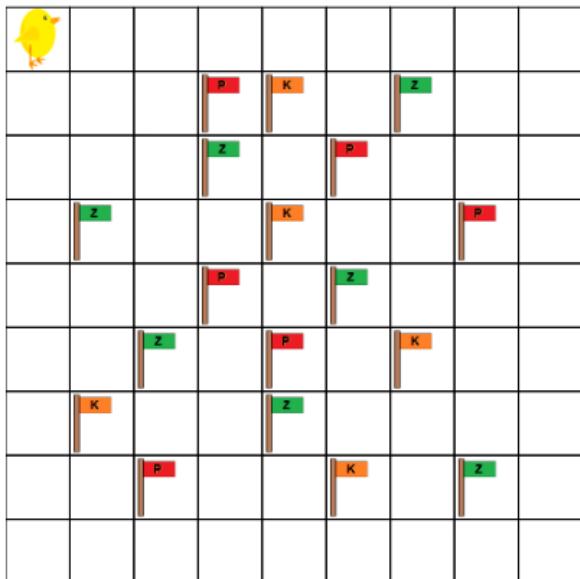
Vsek testni primer ima 6 kupov snega, zato naredimo zanko, ki se ponovi 6 krat. Najprej se premaknemo naprej in poberemo sneg. S senzorjem ugotovimo na kateri strani je hiša in se ustrezno obrnemo, premaknemo naprej, odložimo sneg in se vrnemo nazaj na cesto ter se obrnemo v ustrezno smer.

Povezava do rešitev

ZELENJAVNI VRT

SŠ ZAČETNIKI

Pišek si bo na vrtu posadil zelenjavo. V zemljo je zapičil zastavice z vrsto zelenjave, ki jo bo tam posadil (K – korenje, P – paradižnik, Z – zelje). Sestavi program, ki bo Pišku pomagal posaditi vso zelenjavo. Posadiš jo z ukazom “Posadi”. Zahtevano vrsto zelenjave prebereš s pomočjo senzorjev. Program naj deluje za kakršnokoli razporeditev.



Povezava do naloge

Ideja reševanja

Za rešitev potrebujemo dvojno zanko. Zunanja zanka se ponovi 8 krat, ker je 8 vrstic. Najprej se premaknemo dol in ker imamo v vsaki vrstici 8 stolpcev, se tudi notranja zanka ponovi 8 krat. Premikamo se desno in preverjamo če je polje označeno. V primeru, da je označeno posadimo označeno zelenjavo. Ko zaključimo sprehod po vrstici, se kar po isti vrstici premaknemo nazaj na začetek vrstice (torej imamo še drugi vgnezdeno zanko za premik v levo).

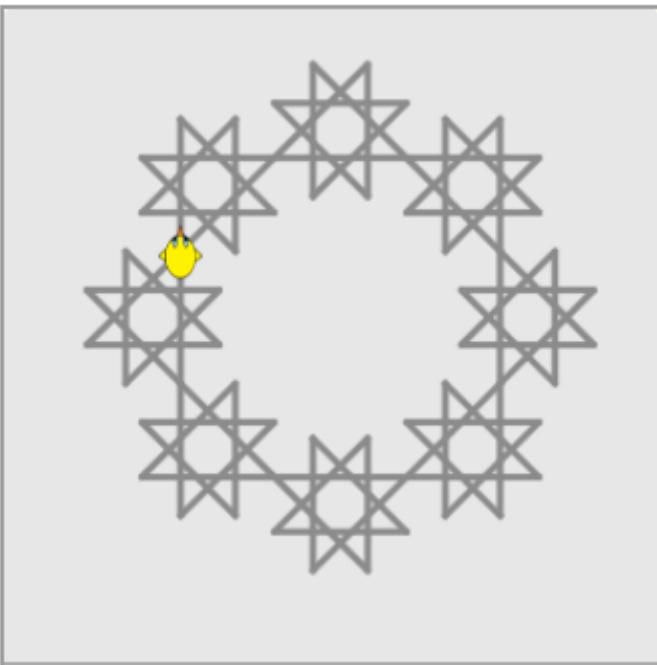
Druga možnost je, da tudi v levo smer sadimo zelenjavo, če je potrebno. Seveda se potem zunanjega zanka ponovi le 4 krat.

Povezava do rešitev

PIŠKOV BOŽIČNI OKRAS

SŠ ZAČETNIKI

Bliža se Božič in Pišek želi okrasiti svoje domovanje. Na tla želi narisati božični okras, sestavljen iz osmih zvezdic. Načrt ima že pripravljen, a mu ga po danih navodilih nikakor ne uspe pravilno narisati. V navodilih (predstavljenih v programu in funkcijah) sta dve napaki. Poišči ju in program popravi tako, da bo Pišku uspelo okrasiti svoj dom po svojih željah.



Povezava do naloge

Ideja reševanja

Sestavljeni imamo že dve funkciji. Funkcija zvezda nariše eno zvezdo v okrasu. Funkcija okras pa s ponavljajočim klicem nariše toliko zvezdic, kot določimo s parametrom stZvezdic. Ker je zvezdic v okrasu 8, mora biti parameter ob klicu funkcije 8, ne 6. Druga napaka je pri zasuku v desno ko se nariše prva zvezdica. Ker je vseh zvezdic 8, mora biti kot zasuka med eno in drugo zvezdico 45° .

Povezava do rešitev

PIŠKOVE VOLITVE

SŠ ZAČETNIKI

V Piškovi šoli so trije razredi. V vseh imajo volitve za predsednika razreda. Za predsednika se v vsakem razredu (pod Test 1, Test 2 in Test 3) potegujeta po dva kandidata. Imeni kandidatov sta zapisani v prvi vrstici. V vsakem razredu je glasovalo 20 učencev. Njihove izbire so napisane v naslednjih vrsticah.

Pomagaj Pišku prešteti glasove in izpiši ime kandidata, ki je zmagal. Če sta bila kandidata izenačena, izpiši »Neodločeno«.

Na delovni površini je že zapisan tisti del programa, ki ime prvega kandidata zapiše v spremenljivko z imenom prvi in ime drugega kandidata v spremenljivko z imenom drugi.

Input: Miha Sara Sara Miha Miha Miha Sara Sara Miha Sara	Output:
Input: Katarina Sabina Sabina Sabina Katarina Sabina Katarina Sabina Sabina Katarina Sabina	Output:
Input: Tine Peter Peter Tine Tine Peter Peter Peter Tine Peter	Output:

Povezava do naloge

Ideja reševanja

Najprej nastavimo dve spremenljivki za štetje glasov na vrednost 0. Med branjem vrstic z dvema spremenljivkama štejemo glasove za prvega oziroma drugega kandidata. To naredimo z zanko za vsak i od 1 do 20 in pri vsakem koraku s pogojnim stavkom preverimo če je prebrano ime enako imenu prvega kandidata. Če je to res, povečamo spremenljivko za štetje glasov prvega kandidata, sicer pa spremenljivko za štetje drugega kandidata.

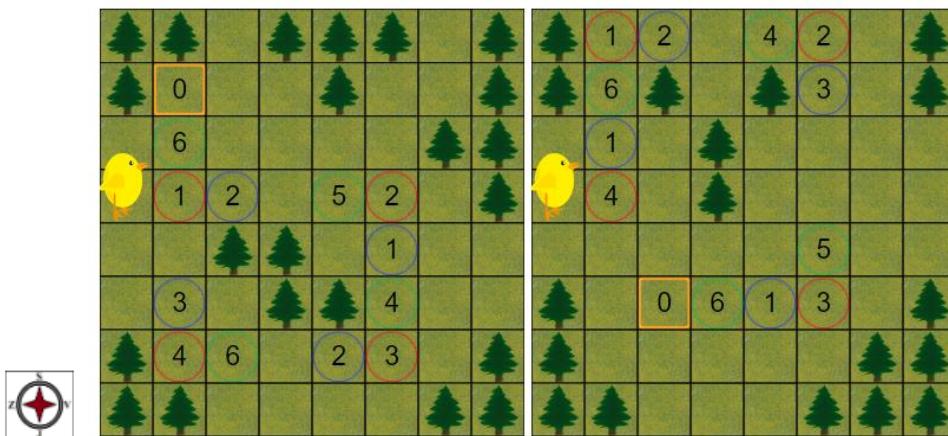
Po zaključku zanke z vgnezdenima pogojnima stavkoma ugotovimo relacijo med njima.

Povezava do rešitev

PIŠEK NA ORIENTACIJSKEM TEKMOVANJU SŠ ZAČETNIKI

Pišek se je udeležil orientacijskega tekmovanja. Ko prispe na prvo polje s številko (obkroženo z rdečo barvo), mu to pove, v katero smer naj se obrne (1 vzhod, 2 jug, 3 zahod, 4 sever). Na naslednjem polju (modro) izve, koliko korakov je do kontrolne točke. Na kontrolni točki (zeleno) dobi nagradne točke. Na naslednjem polju izve novo smer gibanja.

Ko nabere 20 točk ali več, je uspešno opravil nalogu! V zadnjo polje (označeno z 0) zapiši število pridobljenih nagradnih točk.



Povezava do naloge

Ideja reševanja

Najprej se premaknemo naprej do prvega označenega polja. Potem ponavljamo medtem ko je nagrada manjša od 20: s pogojnimi stavki ugotovimo v katero smer se mora Pišek obrniti, in se seveda obrnemo. Naredimo še en korak v pravo smer, preberemo številko iz polja in se premaknemo naprej za toliko korakov kot smo prebrali. Sedaj smo prispeli na polje z nagrado, zato spremenljivki za nagrade prištejemo vrednost iz polja in se premaknemo naprej.

Ko zaključimo zanko, le še v zadnje polje zapišemo vrednost spremenljivke z nagradami.

Povezava do rešitev

OBEŠANJE POSTELJNINE

Pišek je sestavil robota, ki mu pomaga obešati posteljnino. Robot se lahko giblje po dveh vrsticah, razdeljenih na polja tabele. Prva vrstica služi označevanju, druga pa dejanskemu obešanju posteljnine. V prvi vrstici namreč Pišek z znakom X označi, kje je posteljnino treba prijeti s ščipalko. Robot to vrstico pregleda in v drugi vrstici na ustreznem mestu pripne posteljnino.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		X		X	X			X		X	X	X		X	X	
2																

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1			X	X	X		X		X	X		X			X	
2																

Povezava do naloge**Ideja reševanja**

Robot bo pregledal zgornjo vrsto in si ob tem zapomnil, na katerih mestih mora v spodnji vrstici prijeti ščipalko. Za pomnenje je najbolje, da uporabi tabelo. Ko se bo robot vračal, bo v vsakem stolpcu pogledal, če je v tabeli 1 in tam pripel ščipalko.

Povezava do rešitev

ZAPIS URE

Barbara je pripravila spletni vodnik za izlete v hribe. Ko je že vse zapisala, je ugotovila, da ni sledila navodilom, ki predpisujejo, da se ure pišejo s črko h.

Zato sestavi program, ki bo vsako besedo ura, ure, uri in ur v njenem besedilu spremenil v h.

Npr.: Čas hoje: 2 ure -> Čas hoje: 2 h.

Vsaka beseda ima pred sabo znak za presledek, torej v primeru "Murijeva planina, 1 ura", ne sme zamenjati niza "uri" v besedi Murijeva. Predpostavi, da so vse besede za ure zapisane z malimi črkami.

Vhod: Kurji vrh, čas hoje: 3 ure	Izhod:
Vhod: Matajur, čas hoje: 2 ure 15	Izhod:
Vhod: Gradiška Tura, čas hoje: 2 ure	Izhod:
Vhod: Košutnikov Turn, čas hoje: 5 ure	Izhod:
Vhod: Uršlja gora, čas hoje: 1 ura	Izhod:

Povezava do naloge

Ideja reševanja

1. možnost:

Najprej preberemo vrstico z besedilom in ga shranimo v neko spremenljivko. Potem poiščemo, kje se prvič pojavi niz "ura" (ne pozabimo na presledek pred besedo), če te ni, iščemo "ure" in če še tega ni, poiščemo še "uri". V spremenljivko i smo shranili indeks prve črke. Sedaj že lahko sestavimo novo besedilo, kjer zamenjamo ustrezno besedo s črko h. To naredimo z delčkom ustvari besedilo, v treh delih: prvi del od 1. črke do črke na mestu i, drugi del je

februar 2023

vedno črka "h" in tretji del je od ($i+4$). črke (vsak najden niz "ura", "ure" ali "uri" ima 4 znake) do zadnje črke.

Kadar prebrana vrstica nima nobenega od prej iskanih nizov, bo imela spremenljivka i vrednost 0. Zato moramo na koncu še preveriti, če prebrana vrstica mogoče vsebuje niz "ur". V tem primeru sestavimo novo besedilo podobno kot zgoraj, le tretji del je od ($i+3$). črke do konca.

Na koncu izpišemo ustvarjeno novo besedilo.

2. možnost:

Najprej preberemo vrstico z besedilom in ga shranimo v neko spremenljivko.

Z zanko se sprehodimo po indeksih celotnega besedila in vsakič pogledamo če je niz od trenutnega znaka in še dva naprej (torej od i.tega znaka do ($i+2$). znaka) enak nizu "ur" (presledek in "ur").

Če je to res, pogledamo še naslednji znak, če je ta znak mogoče črka "a" ali "e" ali "i". To pogledamo tako, da poiščemo prvo pojavitev ($i+3$). črke v nizu "aei". Če je rezultat tega iskanja več kot 0, lahko ustvarimo novo besedilo na isti način kot pri 1. možnosti. Nato povečamo i za 4, ker preskočimo vse znake v najdenem nizu.

Sicer preverimo, če je ($i+3$). znak mogoče presledek. Če je, ustvarimo novo besedilo z zamenjavo niza "ur" za črko "h" (kot pri 1. možnosti). V tem primeru i povečamo za 3.

V primerih, ko ne najdemo niza "ur", pa i povečamo le za 1.

Na koncu le še izpišemo novo ustvarjeno besedilo.

3. možnost

preberemo besedilo in se s z indeksom i sprehodimo po celotni dolžini besedila. Pogledamo, če je del besedila od indeksa i do i +2 enak "ur". Če je, v novo besedo dodamo h. Sedaj moramo preskočiti ustrezen del besedila, torej i povečati za 2, če pa "ur" sledi še ena od črke "aei" pa še za 1.

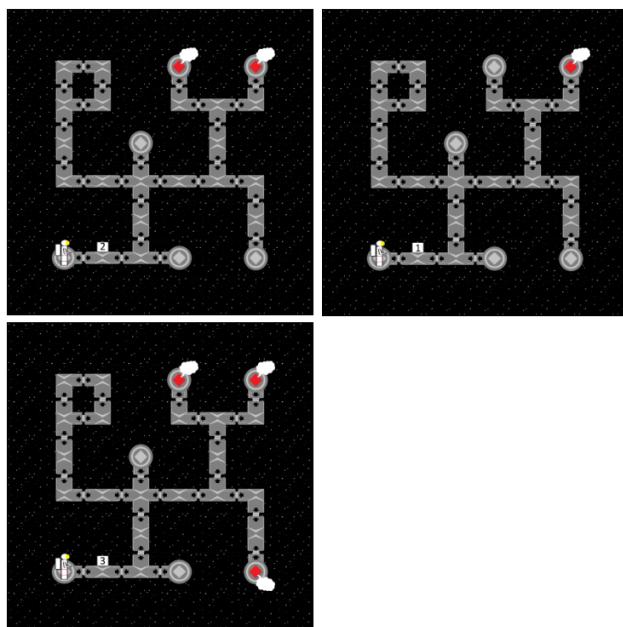
Če pa ustrezeni del ni bil enak "ur". novemu besedilu, dodamo i-to črko besedila.

Povezava do rešitev

KOZMONAVT

Kozmonavt živi na vesoljski postaji. To si lahko predstavljamo kot labirint, sestavljen iz cevi. Vsako jutro mora pregledati in odpraviti vse morebitne okvare. Začne levo spodaj. Na polju pred njim izve število okvar. Te so lahko na mestih, označenimi kot: in so takrat

Pomagaj kozmonavtu in mu pripravi program, s katerim bo vsak dan prišel do vseh okvar in jih odpravil. Pazi! Kot kažejo testi, se napake lahko pojavljajo na različnih mestih pa tudi v različnem številu!



Povezava do naloge

Ideja reševanja

Potem, ko preberemo število okvar, bomo tolikokrat ponovili iskanje okvar. Posamezno okvaro poiščemo tako, da se vedno najprej poskusimo premakniti levo, če ne gre, gremo naravnost. Če še tja ne gre, nadaljujemo desno. Na ta način bomo pregledali celotno vesoljsko postajo. Ko smo na okvari, jo popravimo.

Povezava do rešitev

NEZADOSTNA ŠTEVILA

Pišek se je v šoli učil o različnih vrstah števil. Med njimi so ga najbolj pritegnila števila, ki jim rečemo nezadostna. Nezadostna števila so tista, katerih vsota deliteljev (vključno s samim številom) je manjša od dvakratnika števila. Število 12 ni nezadostno število. Njegovi delitelji so 1, 2, 3, 4, 6 in 12, kar je skupaj 28. Število 15 pa je nezadostno, saj se njegovi delitelji 1, 3, 5 in 15 seštejejo v 24, kar je manj kot 30.

Pišek je poizkušal napisati program, ki bi preveril, ali je vneseno število nezadostno. Pri tem je naredil nekaj napak. Popravi njegov program.

Input: 6	Output:
Input: 11	Output:
Input: 23	Output:
Input: 35	Output:
Input: 18	Output:

Povezava do naloge

Ideja reševanja

V glavnem programu opazimo, da je pogoj za nezadostnost števila napisan narobe.

V funkciji moramo v zanki iti vse do števila, v sami zanki pa vsoti prištetih deli število.

Povezava do rešitev

PIŠEK GLEDA TEKMO

Pišek je zamudil na košarkarsko tekmo. Prišel je čisto na koncu in videl, da je njegova ekipa dosegla 4 koše. Prijatelj mu je začel navdušeno razlagati:

»Veš, najprej smo dosegli 1 točko iz prostih, potem pa smo zadeli še za 2 in potem še za 1.«

»Ne, ne, ni bilo tako ... Najprej smo zadeli za 2, potem pa dvakrat za 1.«

»Nimate pojma,« se je oglasil tretji ... »Začeli smo z zadetim prostim metom, potem pa smo dali trojko.«

Pišek je zdaj čisto zmeden. Zato se je določil, da si bo sestavil program, ki mu bo za dano število točk povedal, na koliko načinov bi lahko ekipa dosegla toliko točk. Tako je ugotovil, da so njegovi prijatelji lahko dosegli 4 točke na kar 7 različnih načinov (če slučajno ne veš, pri košarki lahko z enim košem dosežemo 1, 2 ali 3 točke):

$$4 = 1 + 1 + 1 + 1$$

$$4 = 1 + 1 + 2$$

$$4 = 1 + 2 + 1$$

$$4 = 2 + 1 + 1$$

$$4 = 1 + 3$$

$$4 = 2 + 2$$

$$4 = 3 + 1$$

Sestavi torej funkcijo koši s parametrom točke, ki vrne število načinov, na katere lahko ekipa pri košarki doseže točke točk!

Vhod: 2	Izhod:
Vhod: 3	Izhod:
Vhod: 4	Izhod:
Vhod: 10	Izhod:

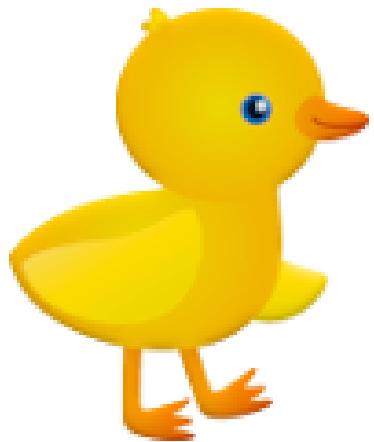
Povezava do naloge

Ideja reševanja

Denimo, da na semaforju piše, da je ekipa dosegla 15 točk. V zadnjem metu je lahko dosegla 1, 2 ali 3 točke. Če je bil zadnji met za 1 točko je torej tekma potekala na toliko načinov, kot se da doseči 14 točk. Če pa je bil zadnji met vreden 2 točki, je bilo načinov, da pridemo do 15 točk (ob zadnjem metu za 2) toliko, kot je bilo načinov, da pridemo do 13 točk. Podobno, če je bil zadnji met trojka.

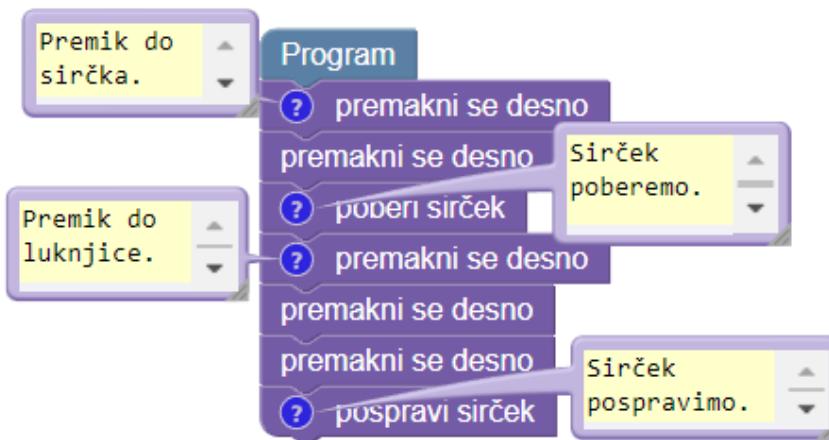
In ker ne vemo, kakšen je bil zadnji met, moramo torej seštetiti načine, da pridemo do 14, 13 in 12 točk. Tukaj pa gre za povsem enak problem, torej uporabimo rekurzijo - funkcija bo poklicala samo sebe (v našem primeru kar 3x)

Povezava do rešitev

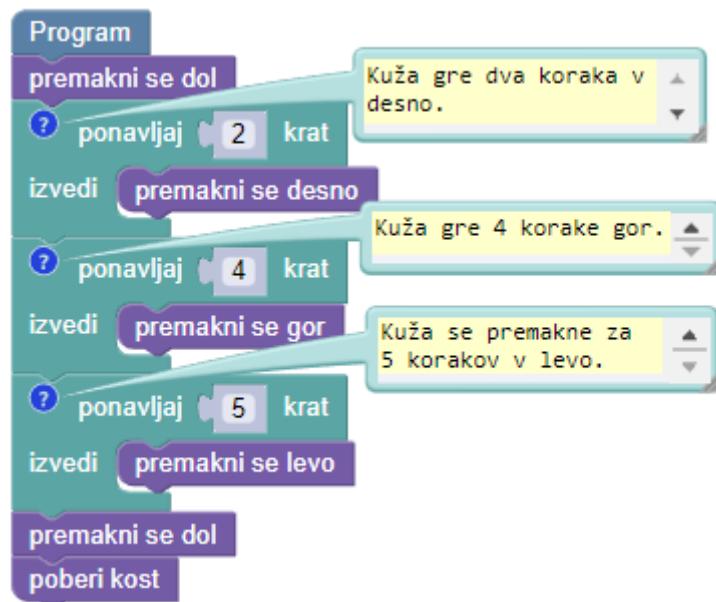


Rešitve

Miška Mimi in njen sirček



Pes išče kost



Prvo zanko seveda lahko zamenjamo z dvema premikoma v desno.



Runo se seveda lahko odpravi tudi po drugi poti. Recimo takole



ali pa

Program

```
premakni se dol
premakni se desno
premakni se desno
ponavljaj [4 krat]
  izvedi premakni se gor
ponavljaj [4 krat]
  izvedi premakni se levo
premakni se dol
premakni se levo
poberi kost
```

Program

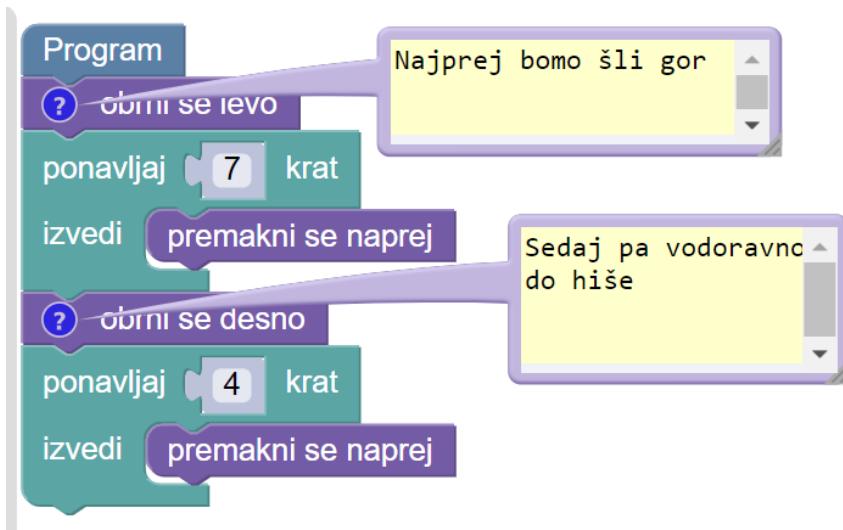
```
premakni se dol
premakni se desno
premakni se desno
ponavljaj [4 krat]
  izvedi premakni se gor
ponavljaj [4 krat]
  izvedi premakni se levo
premakni se levo
premakni se dol
poberi kost
```

Dostava pice

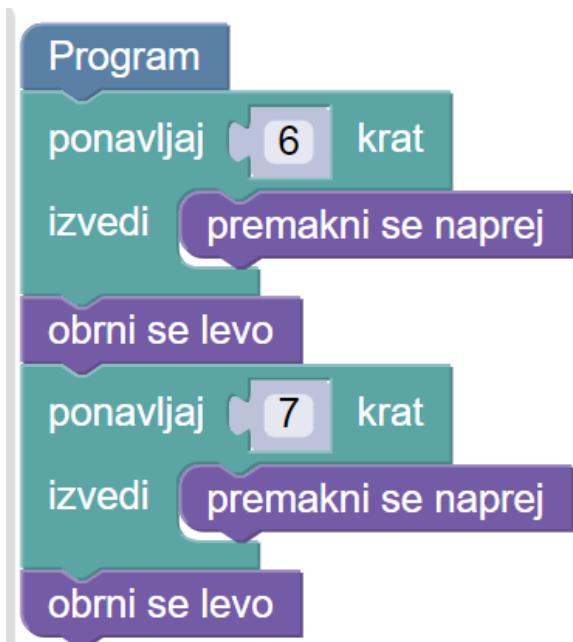
Program

```
ponavljaj [4 krat]
  izvedi premakni se naprej
  obrni se levo
ponavljaj [7 krat]
  izvedi premakni se naprej
```

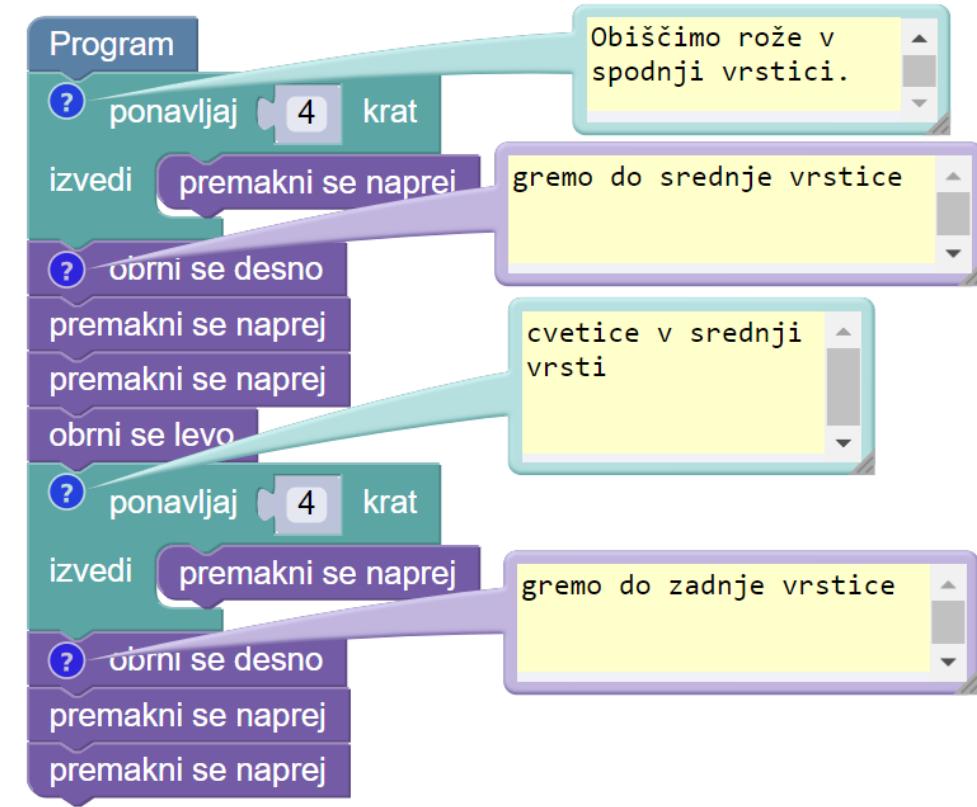
Poti je seveda več. Recimo še ta



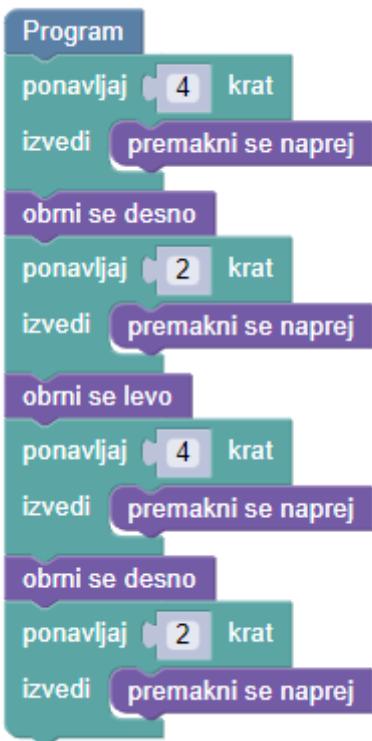
A moraš paziti. Če bo dostavljalec preveč taval, mu bo zmanjkalo delčkov. Kot npr. tukaj, ko na voljo nima več delčkov, pa še ni čisto pri hiši:



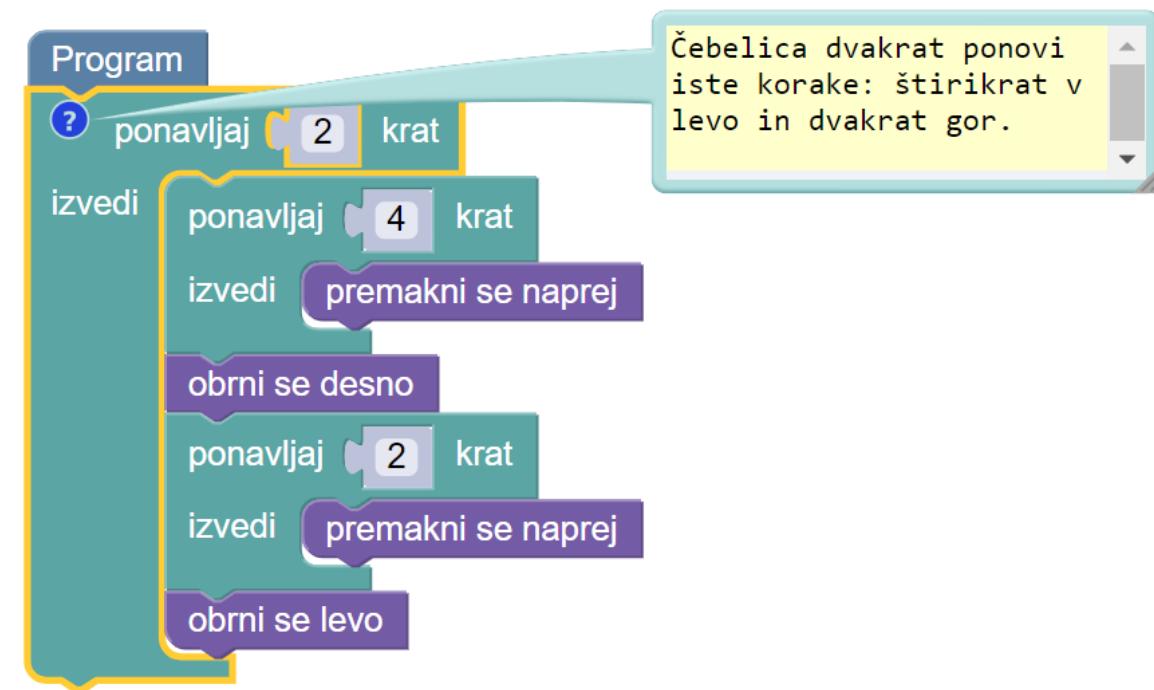
Čebelica leti s cveta na cvet



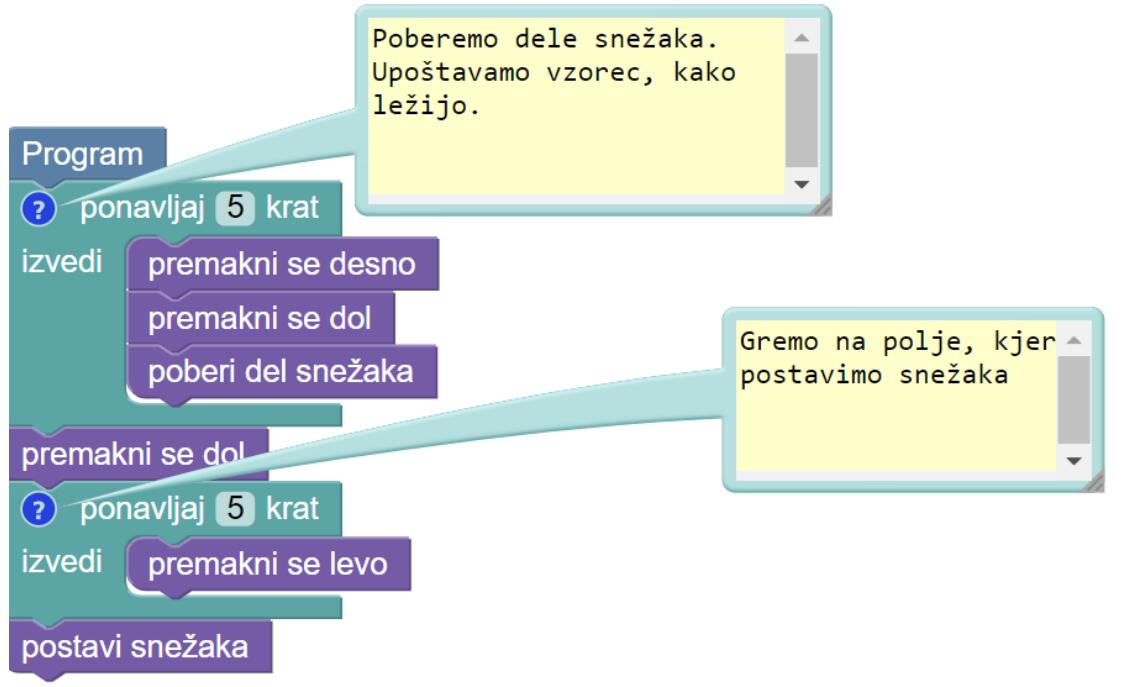
Ta program lahko zapišemo tudi malo drugače:



Lahko pa sledimo ideji, da opazimo vzorec, ki se dvakrat ponovi in uporabimo gnezdeno zanko:



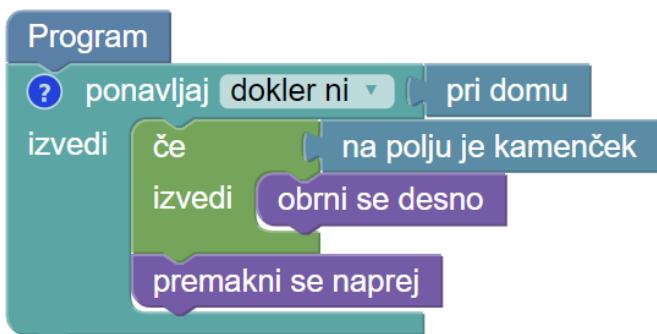
Peter naredi snežaka



Janko in Metka



ali pa zamenjamo gibanje in obračanje



Metulj Teo riše

Program

```
pojdi naprej za 30
zavij levo ⌂
pojdi naprej za 10
zavij desno ⌂
pojdi naprej za 10
zavij desno ⌂
pojdi naprej za 30
zavij desno ⌂
pojdi naprej za 10
zavij desno ⌂
pojdi naprej za 10
zavij levo ⌂
pojdi naprej za 30
zavij desno ⌂
pojdi naprej za 10
```

Lačna žabica

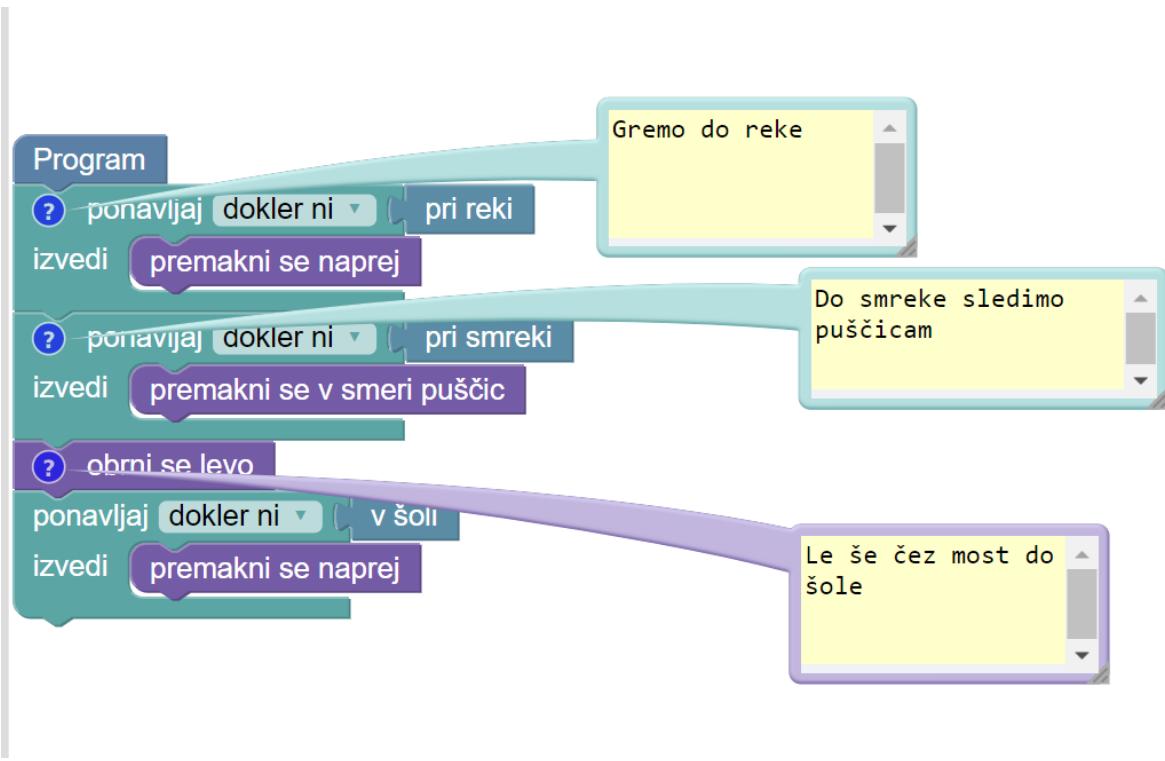
Program

```
? ponavljaj 4 krat
izvedi ponavljam 3 krat
izvedi skoči naprej
? če muha na lokvanju
izvedi pojem muha
obrni se desno
```

Uporabimo dvojno zanko:
z notranjo zanko žabica skače po lokvanjih.
Zunanja zanka je za vsako stranico kvadrata

Če je na lokvanjevem listu muha, jo bo žabica pojedla.

Gregor išče šolo



Kekec v labirintu



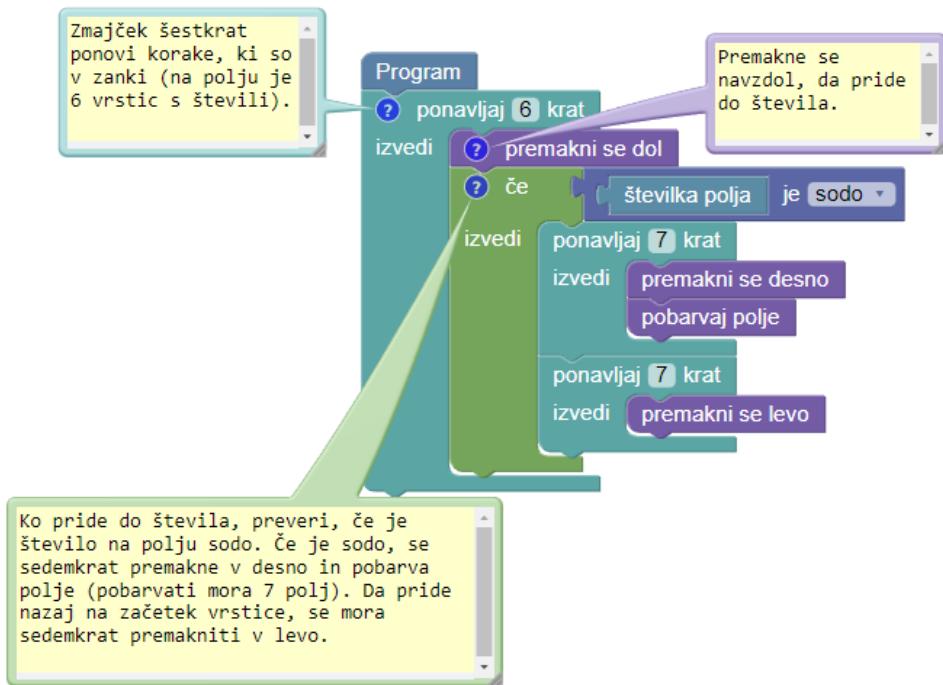
ali pa



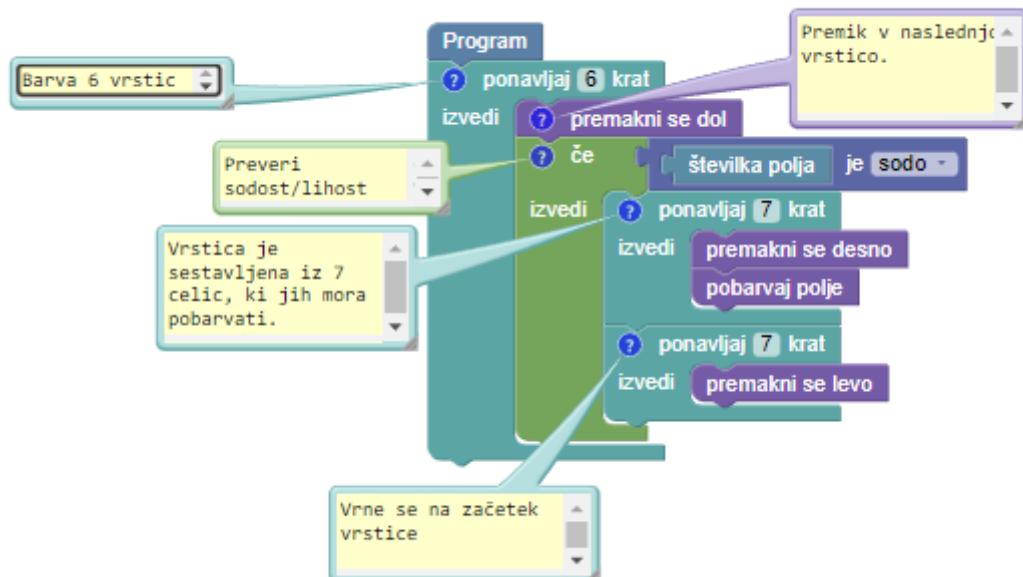
ali pa



Preberi in pobarvaj



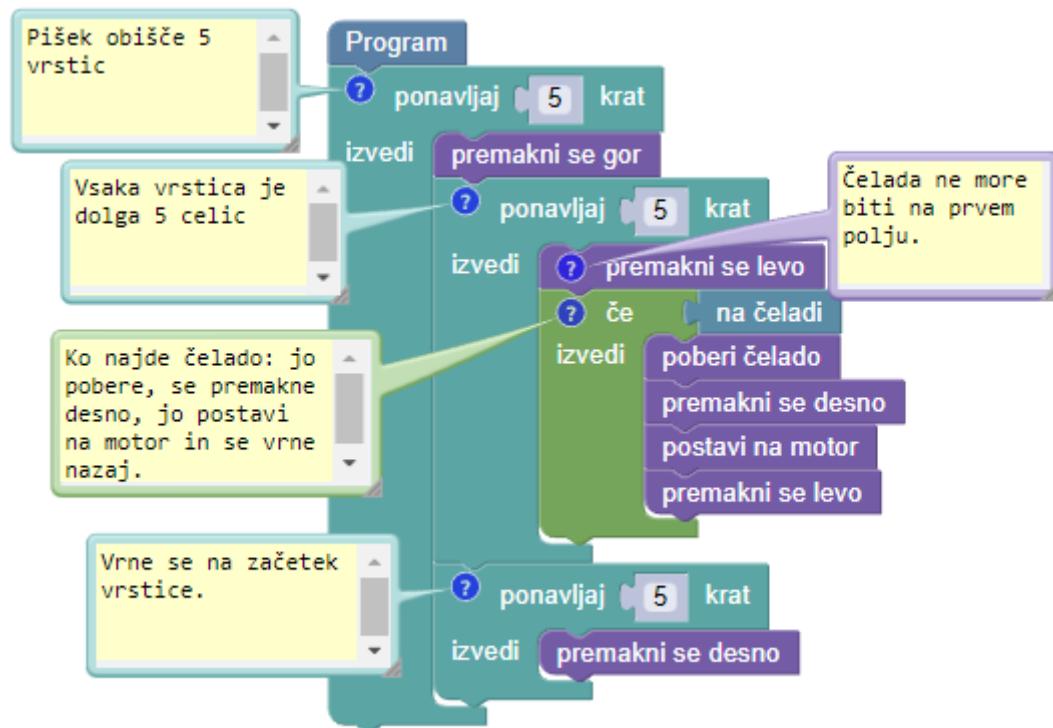
Ko pride do števila, preveri, če je število na polju sodo. Če je sodo, se sedemkrat premakne v desno in pobarva polje (pobarvati mora 7 polj). Da pride nazaj na začetek vrstice, se mora sedemkrat premakniti v levo.

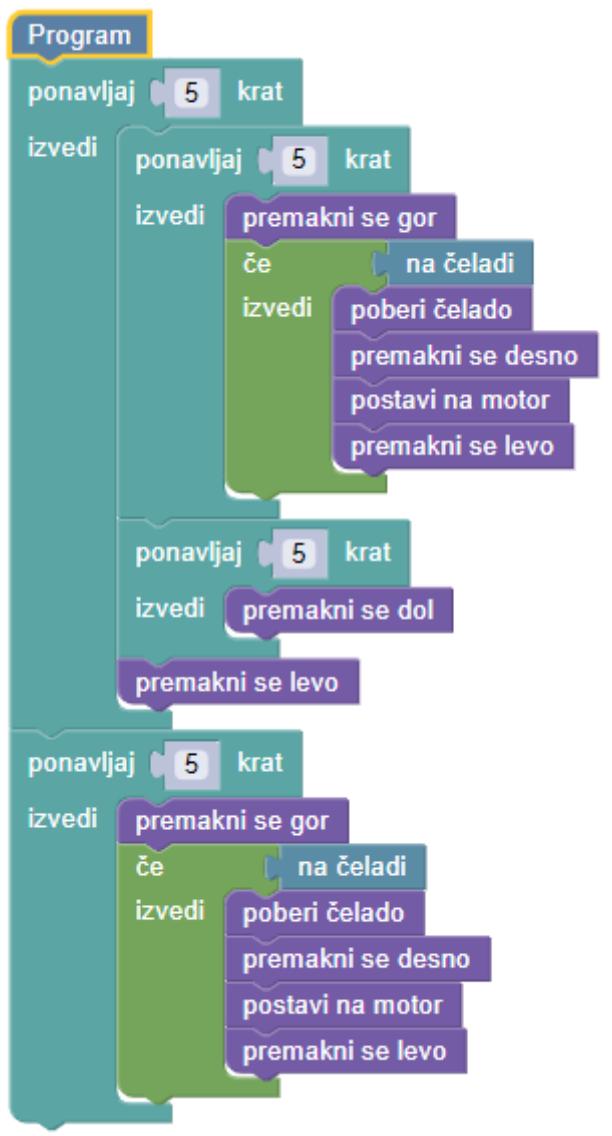


Vrstice je sestavljena iz 7 celic, ki jih mora pobarvati.

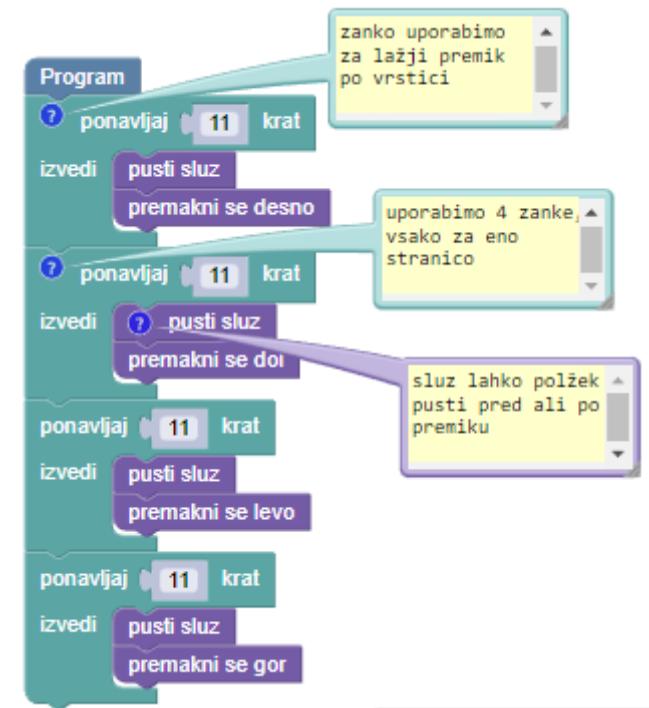
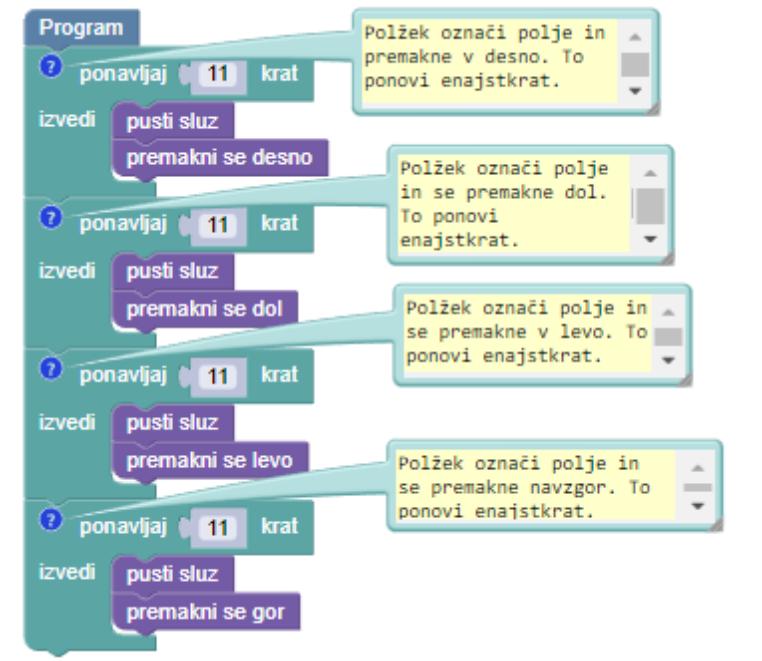
Vrne se na začetek vrstice

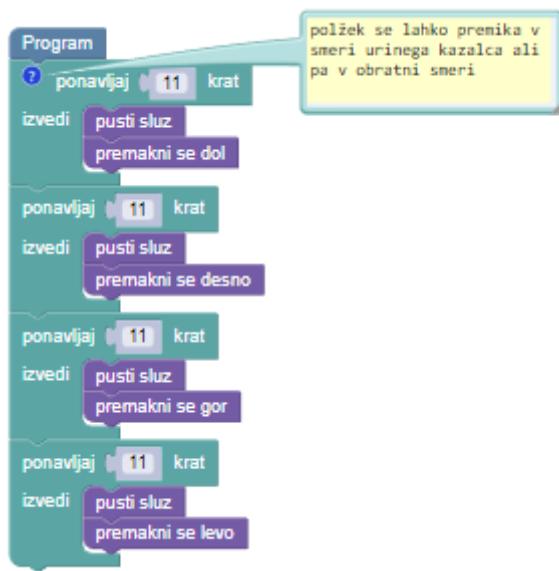
Motoristi



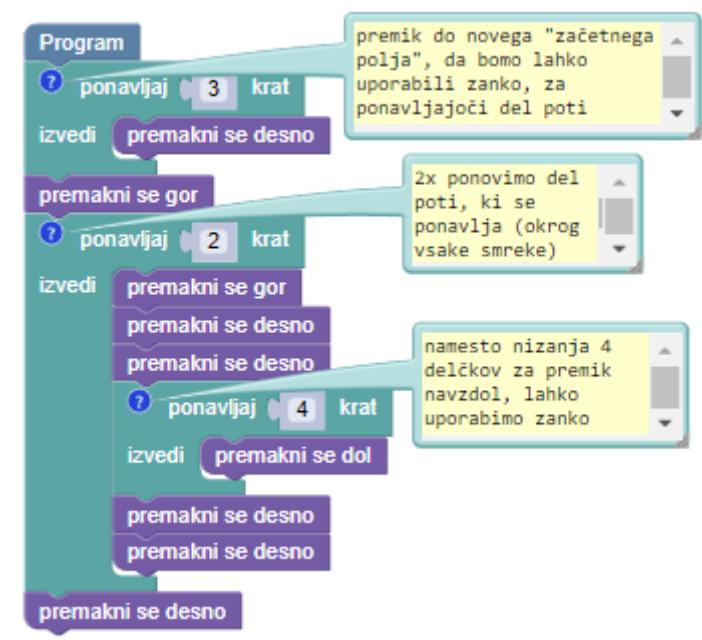


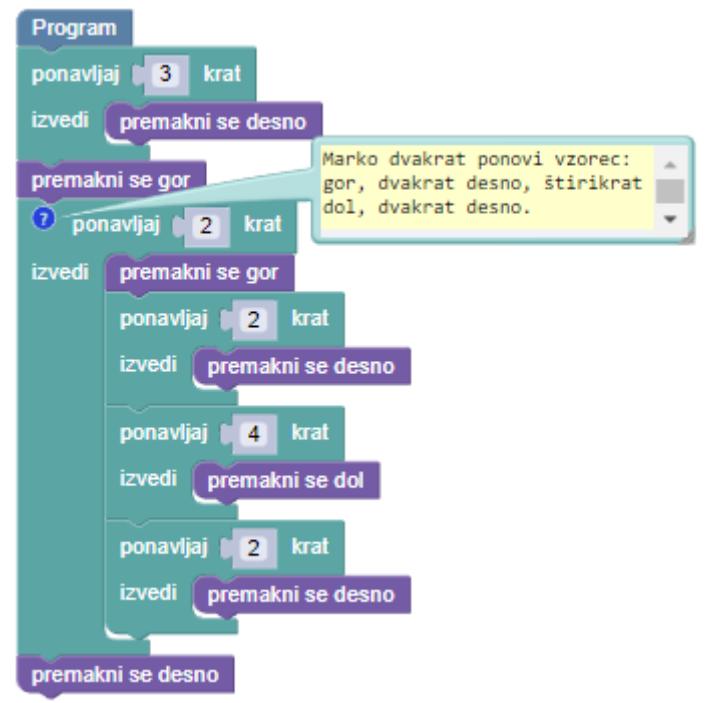
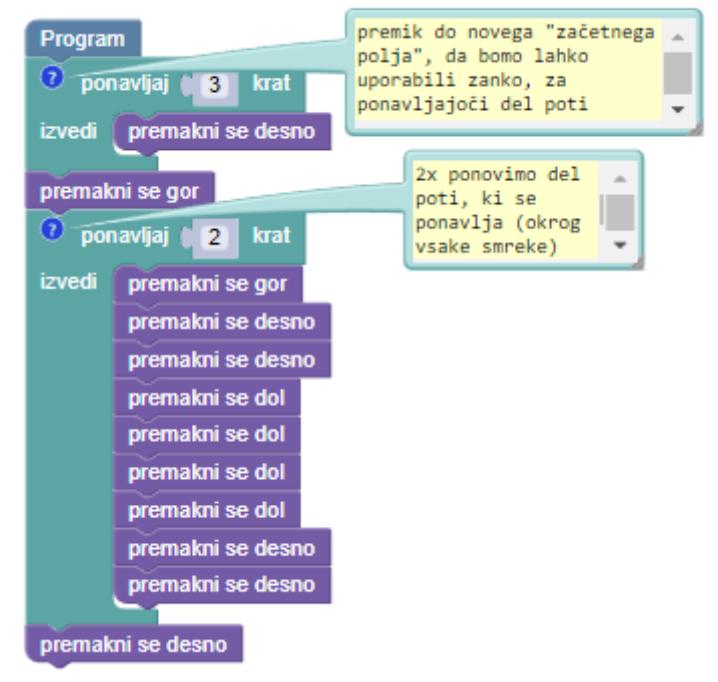
Polžek Slinček





Tačko in skrivalnice





Program

```
premakni se desno
premakni se dol
premakni se dol
premakni se desno
premakni se desno
premakni se gor
premakni se gor
premakni se desno
premakni se desno
ponavljam [4] krat
izvedi premakni se dol
1
premakni se desno
premakni se desno
premakni se gor
premakni se desno
premakni se desno
ponavljam [4] krat
izvedi premakni se dol
premakni se desno
premakni se desno
premakni se desno
```

Janko in Metka

Program

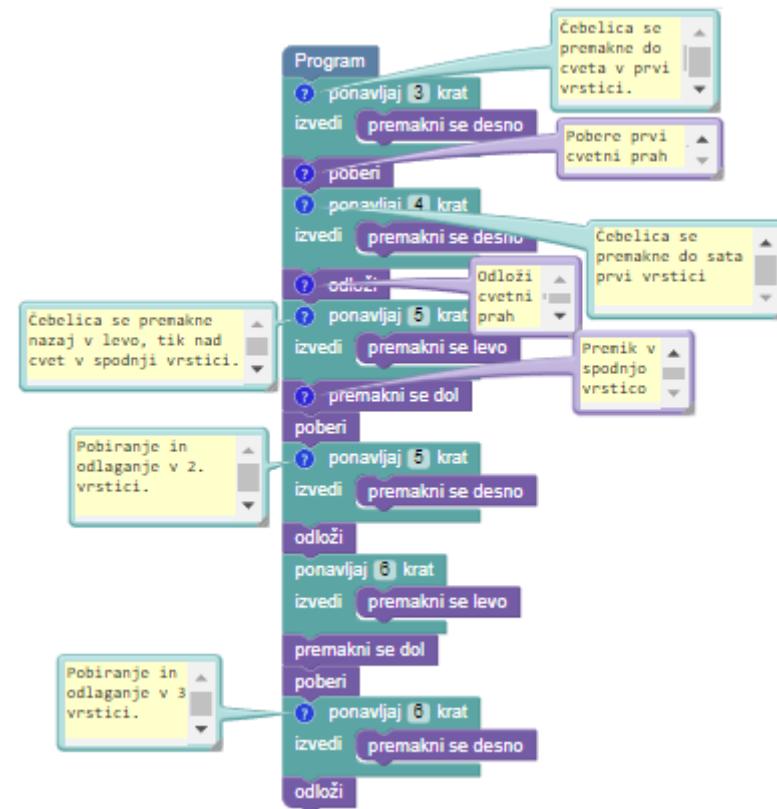
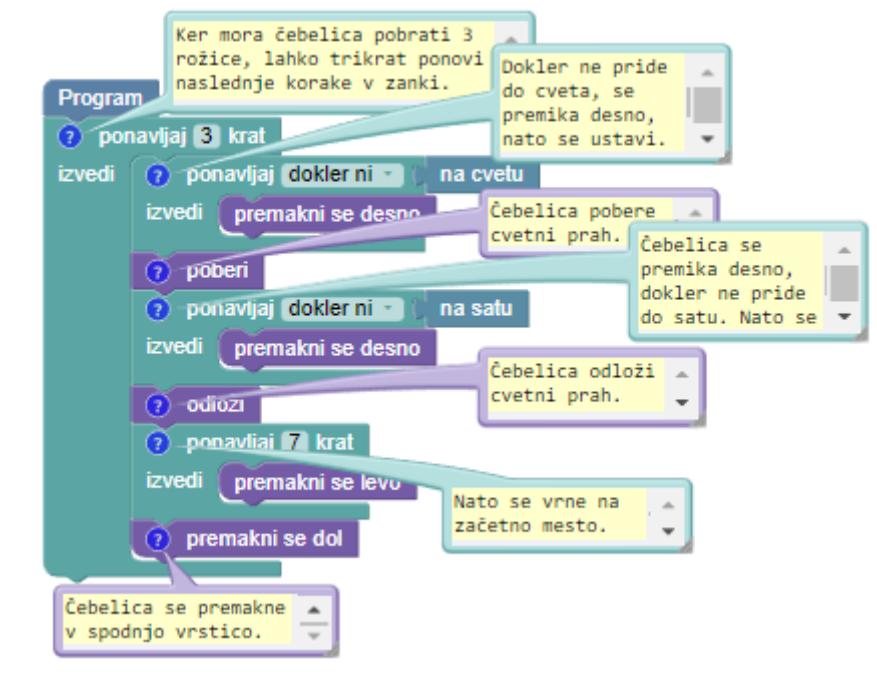
```
? ponavljam dokler ni [pri domu]
izvedi ? premakni se naprej
? če na polju je kamenček
izvedi obmi se desno
```

Janko in Metka se premikata po polju, dokler ne prideta do hiške.

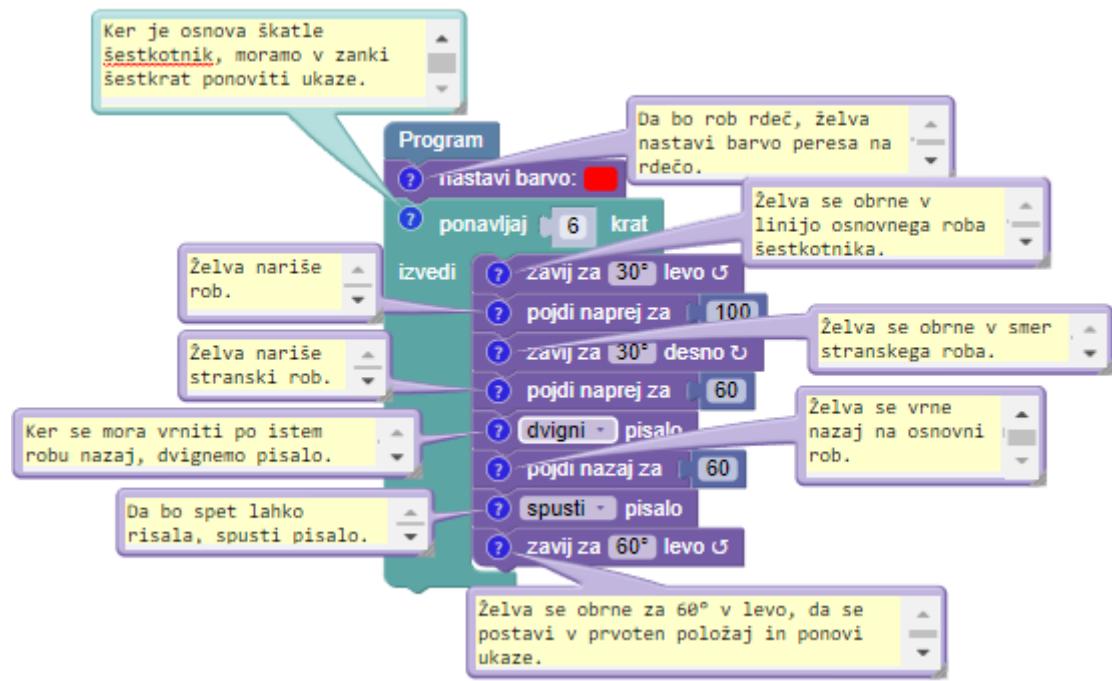
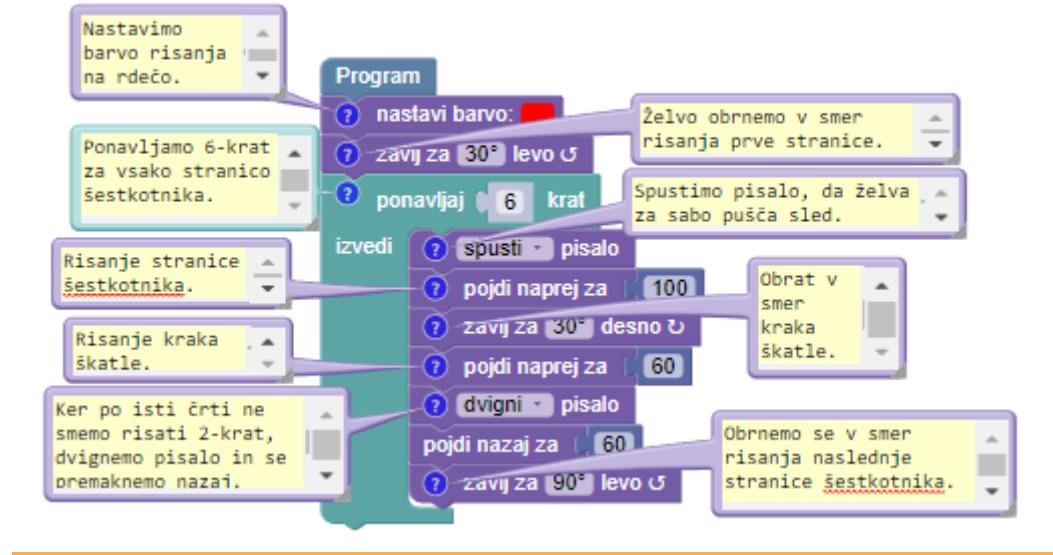
Premikata se naprej. Kadar prideta na polje s kamenčkom preverita pogoj.

Če je na polju kamenček, se obrneta desno.
Če pa pogoj ni izpolnjen, gresta ponovno naprej.

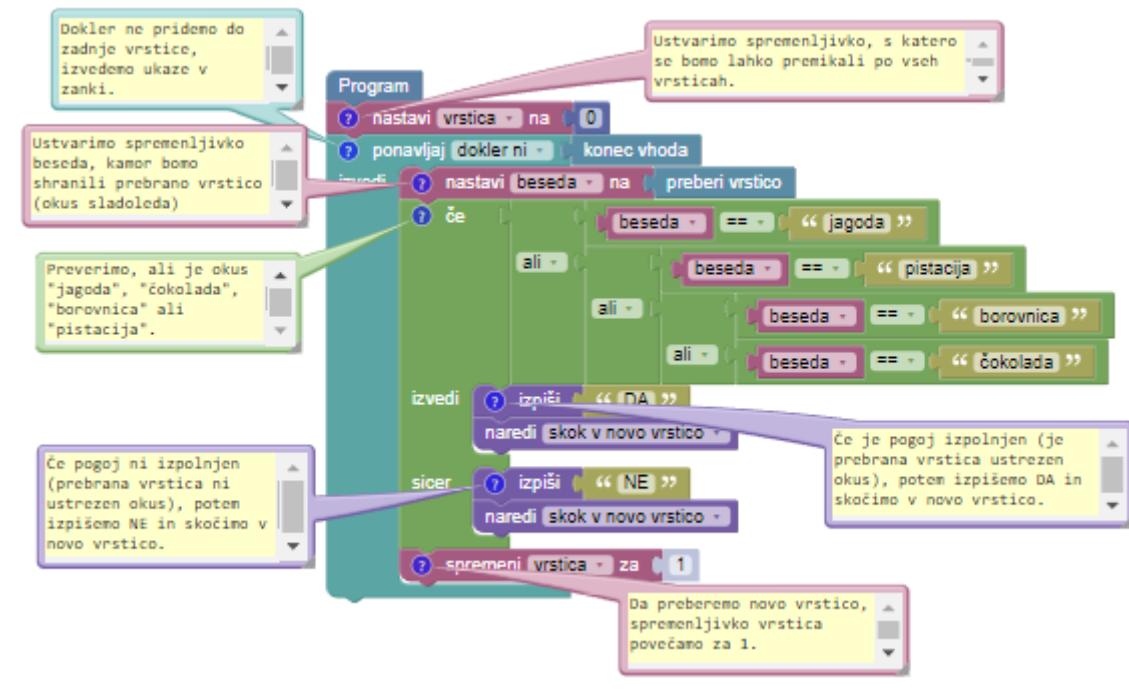
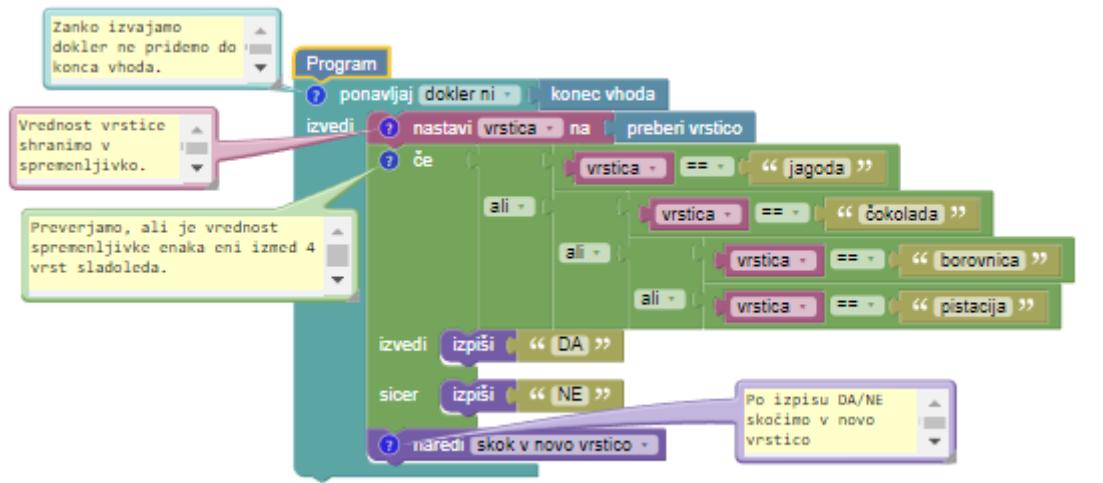
Čebelica Maja nabira cvetni prah



Škatla za semena



Sladoled



```

Zanko izvajamo,
dokler ne pridemo do
konca vhoda.

Vrednost, ki jo
preberemo v vrstici,
shranimo v
spremenljivko.

Izpišemo
DA oz. NE

Po izpisu DA/NE
skočimo v novo
vrstico

Preverjamo, ali je vrednost
spremenljivke enaka okusu

```

Program

ponavljam dokler ni konec vhoda

izvedi nastavi vrstica na proba

če vrstica == "jagoda"

izvedi izpiši "DA"

če vrstica == "vanilija"

izvedi izpiši "NE"

če vrstica == "čokolada"

izvedi izpiši "DA"

če vrstica == "pistacija"

izvedi izpiši "DA"

če vrstica == "kokos"

izvedi izpiši "NE"

naredi skok v novo vrstico

Čebelica leti s cveta na cvet

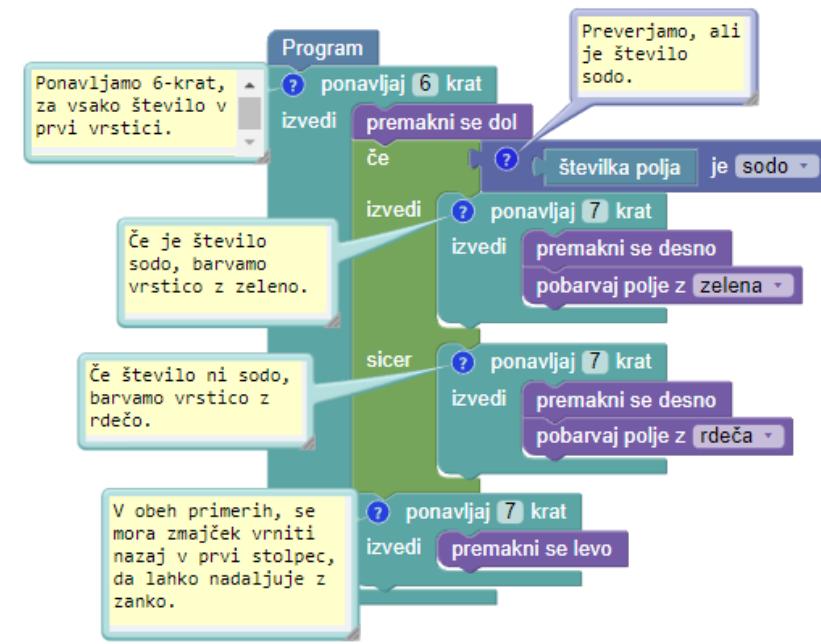
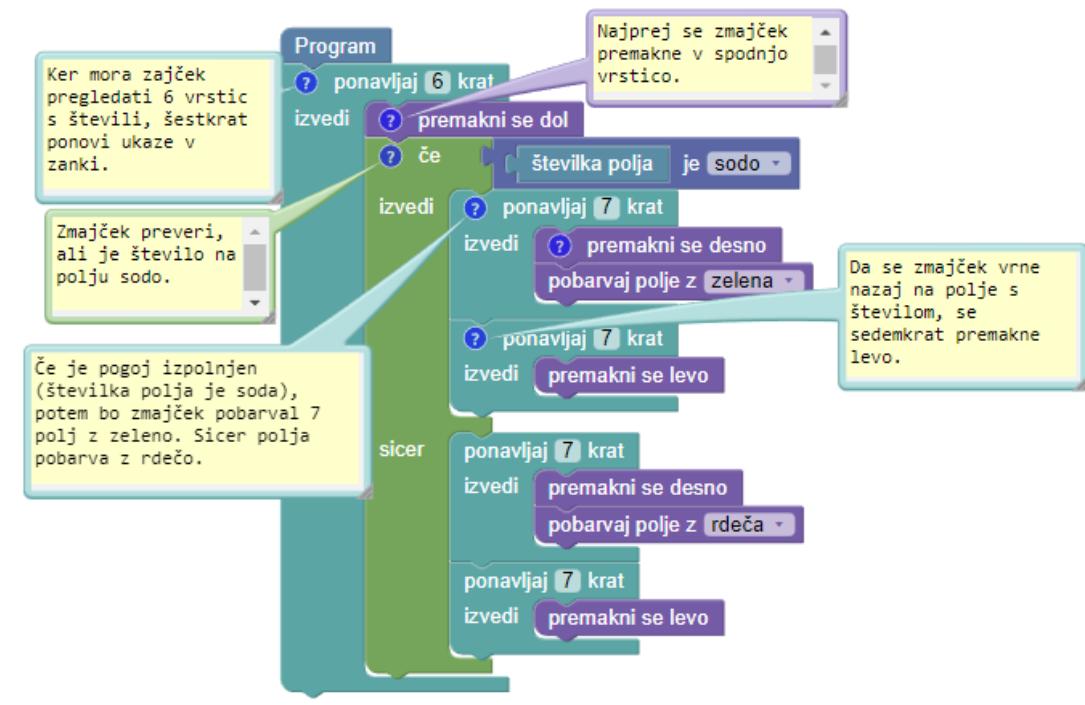
```

Čebelica dvakrat ponovi
iste korake: štirikrat v
levo in dvakrat gor.

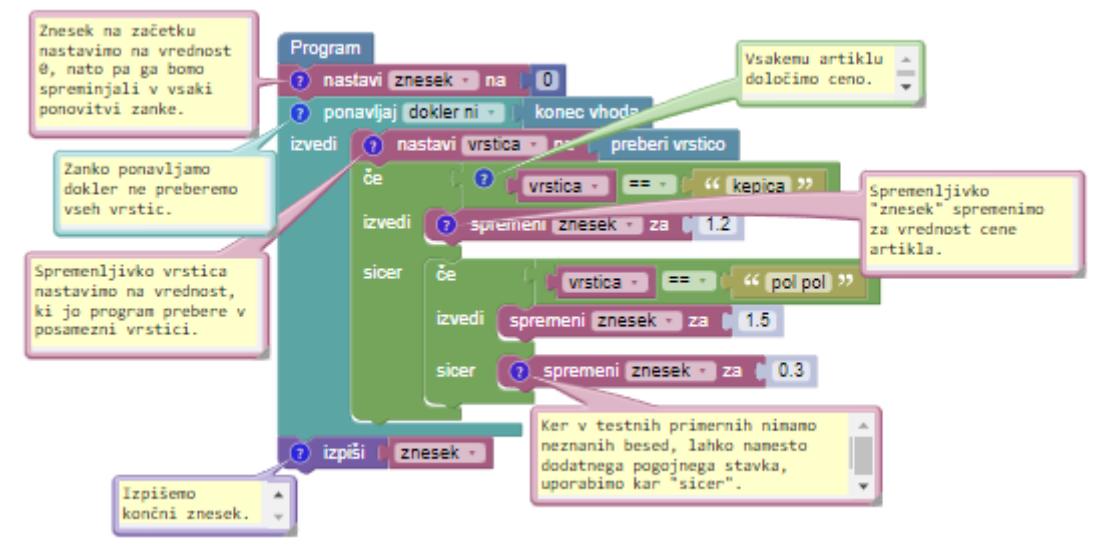
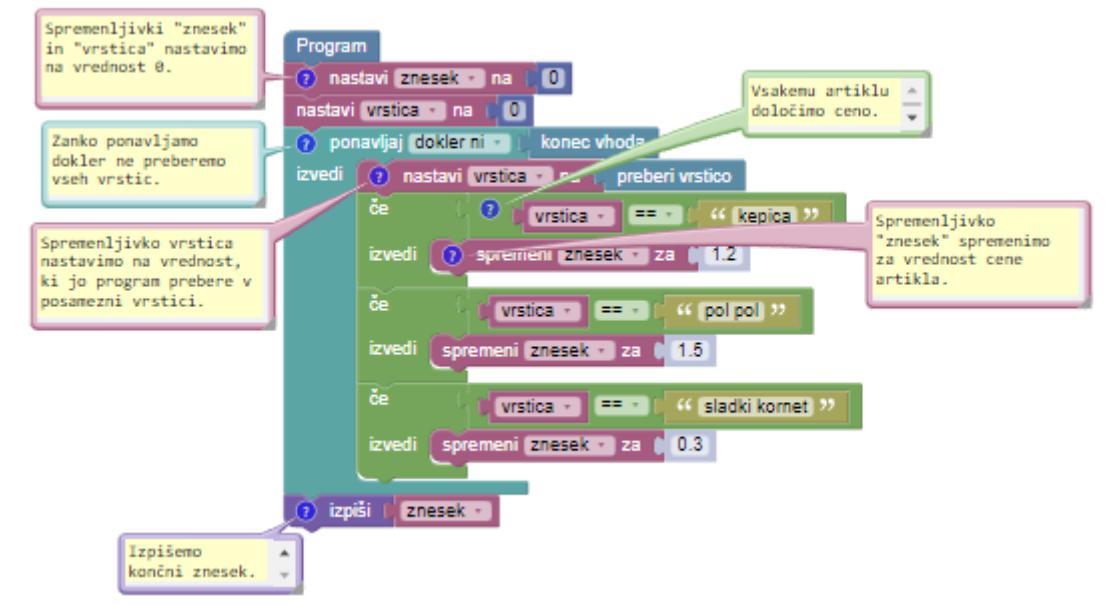
ponavljam 2 krat
izvedi ponavljam 4 krat
izvedi premakni se naprej
obrni se desno
ponavljam 2 krat
izvedi premakni se naprej
obrni se levo

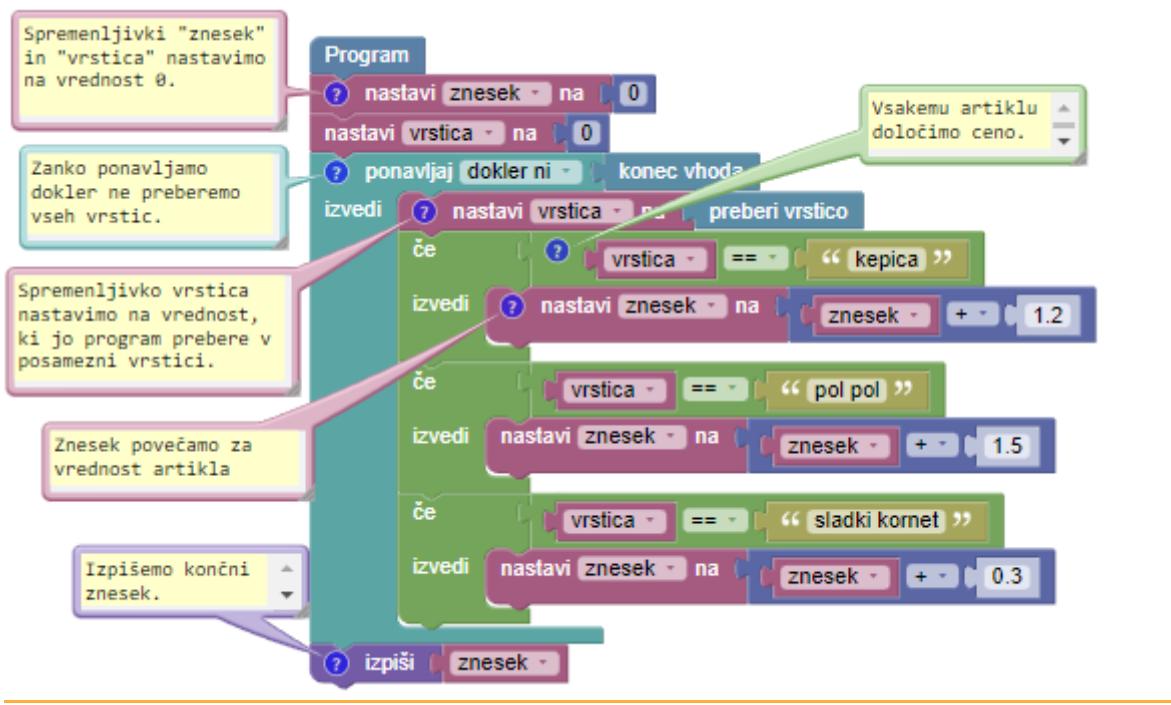
```

Preberi in pobarvaj

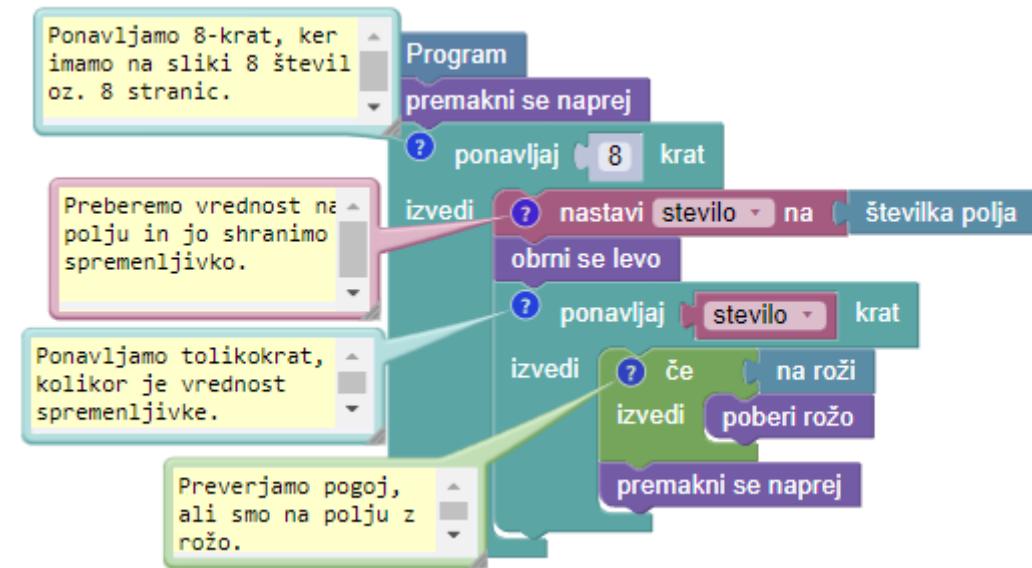


Sladoled



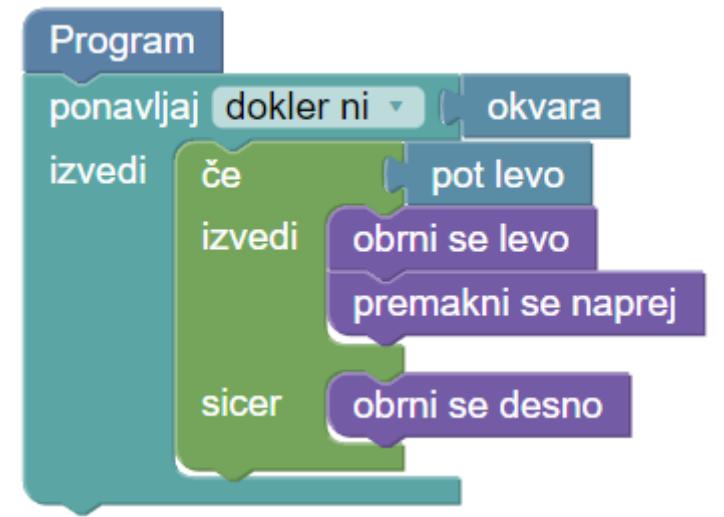
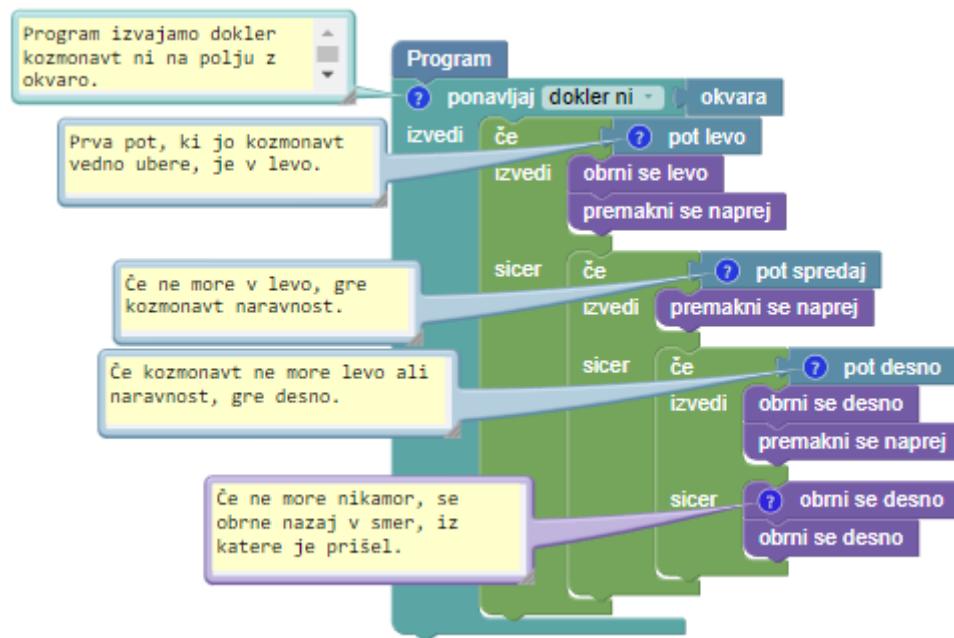


Kekec v labirintu

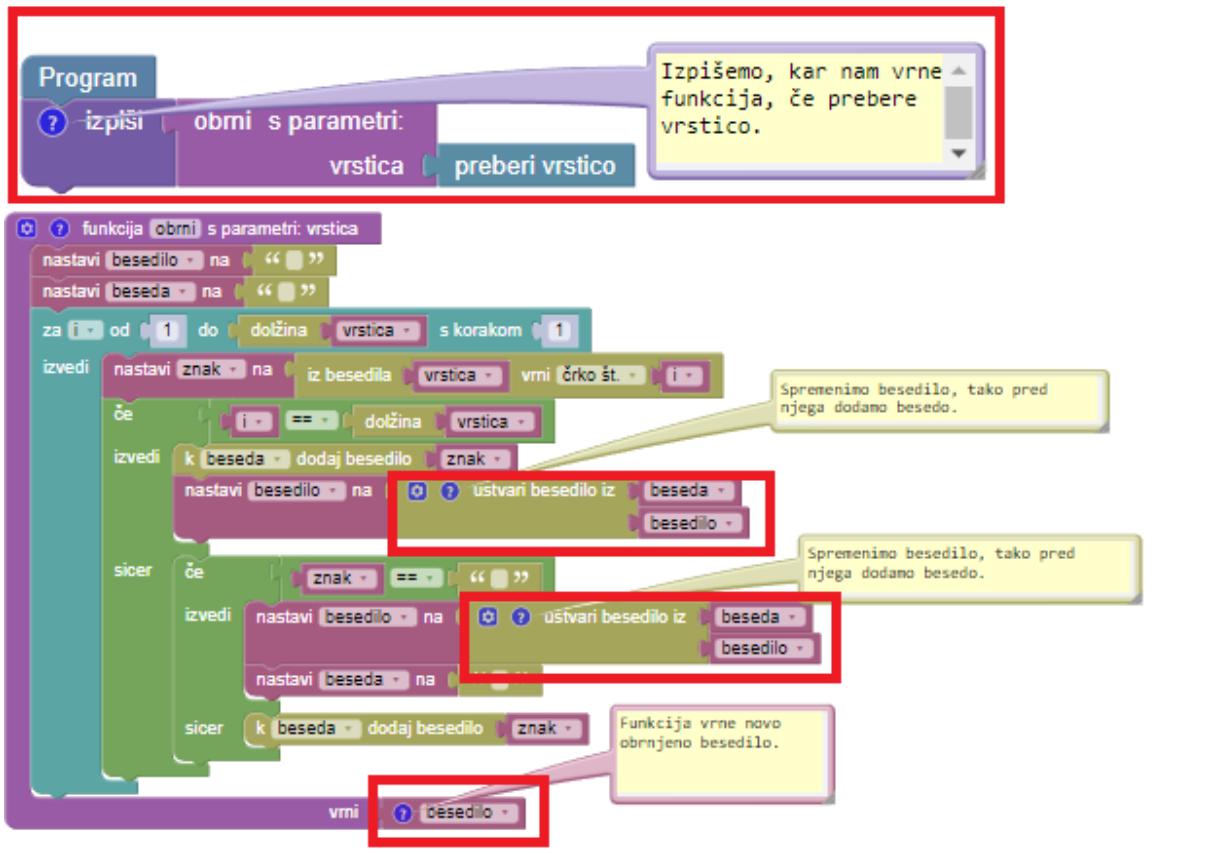




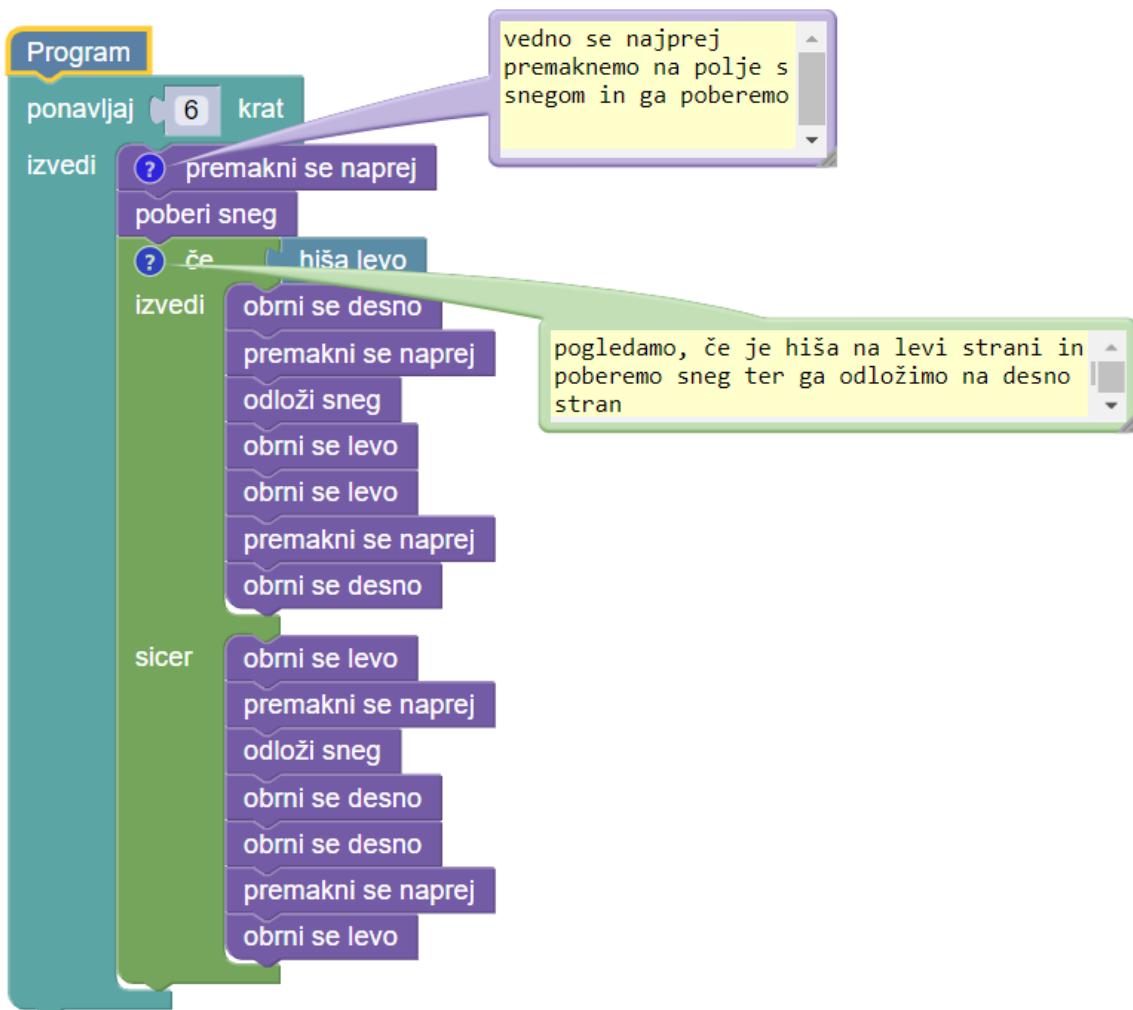
Kozmonavt

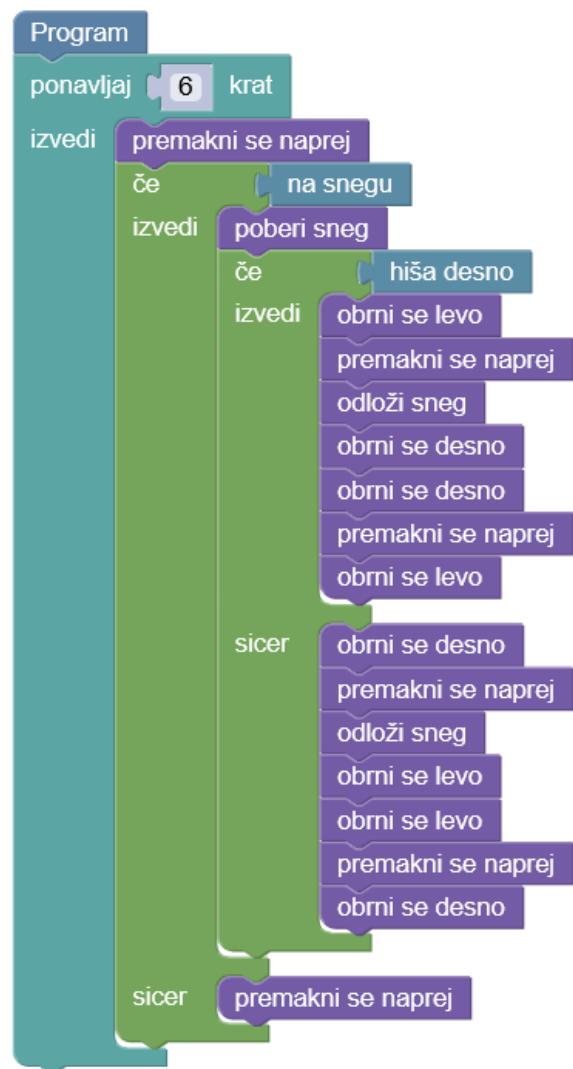


Izbira gesla

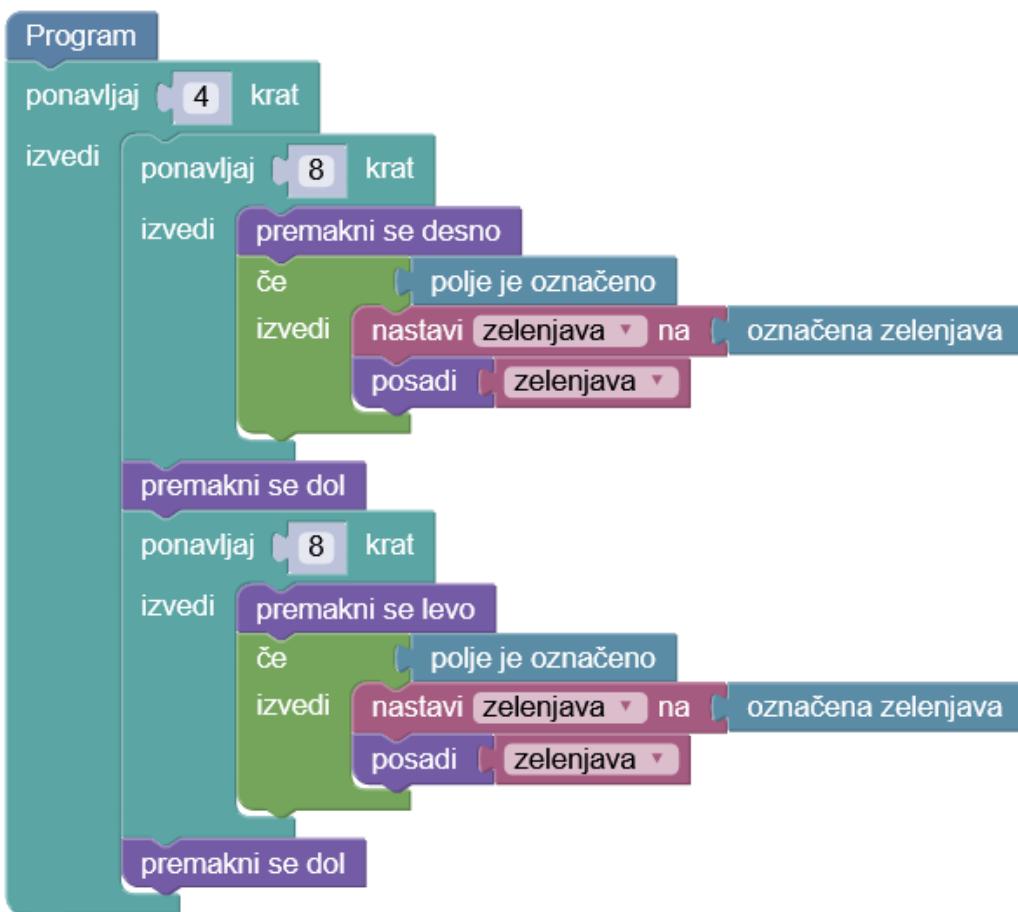
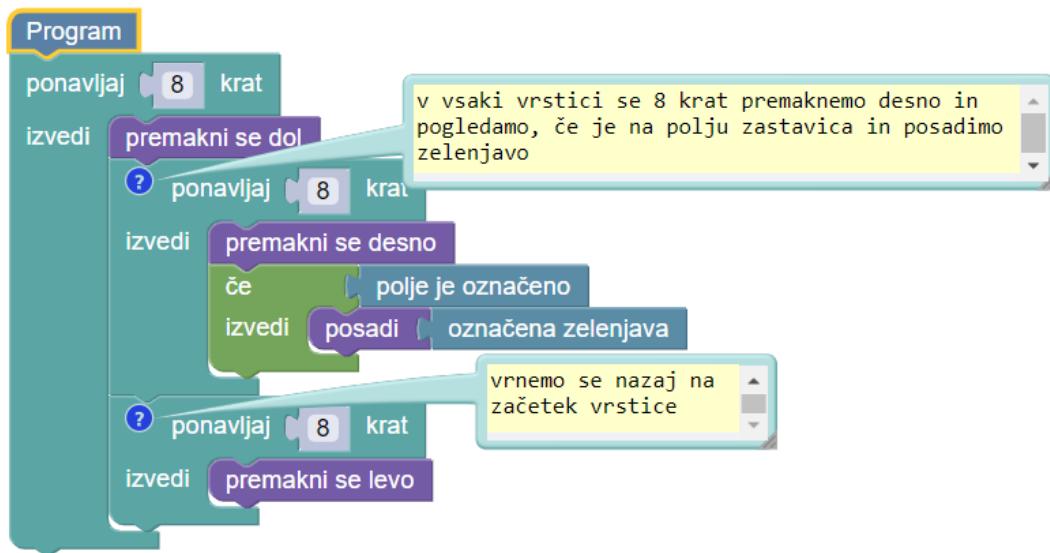


Zasnežena cesta

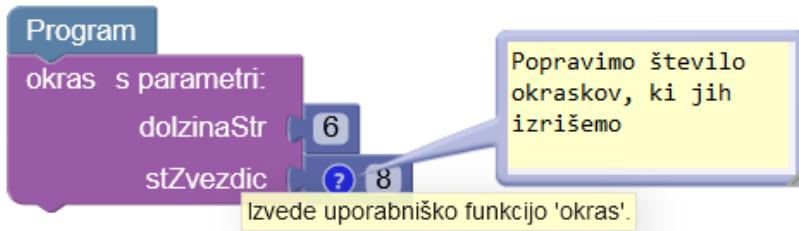




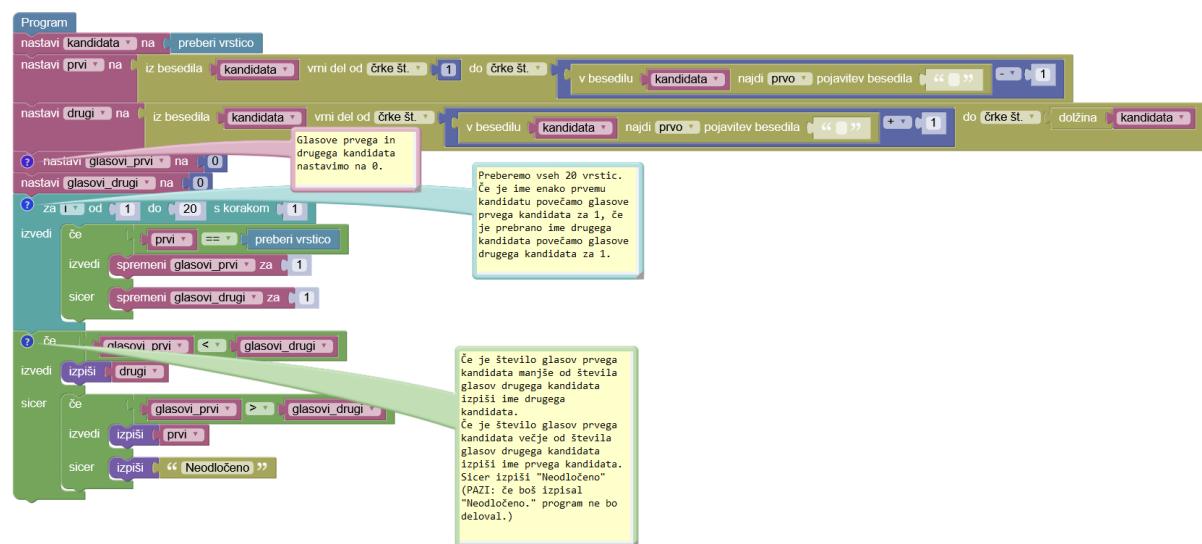
Zelenjavni vrt



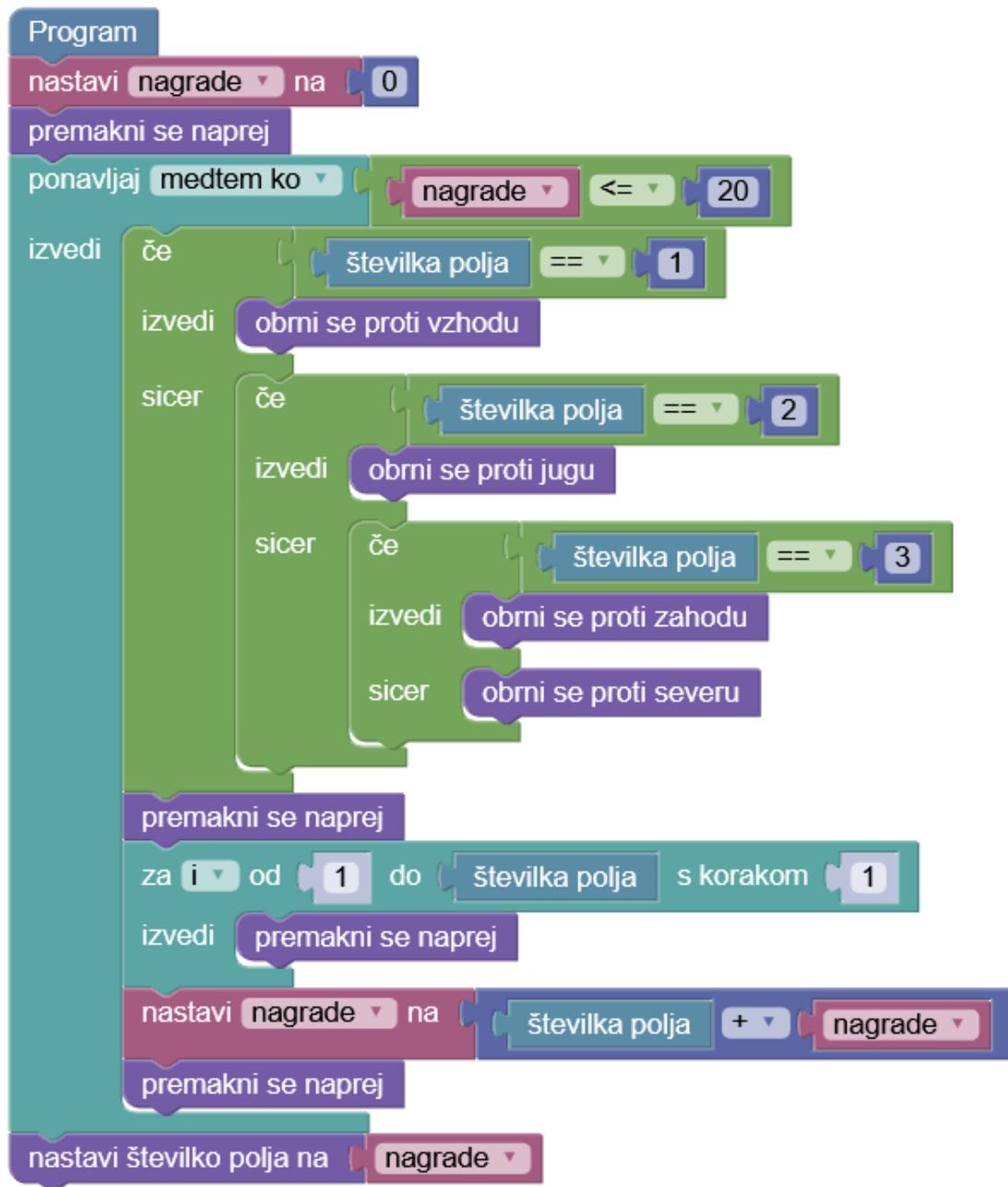
Piškov božični okras



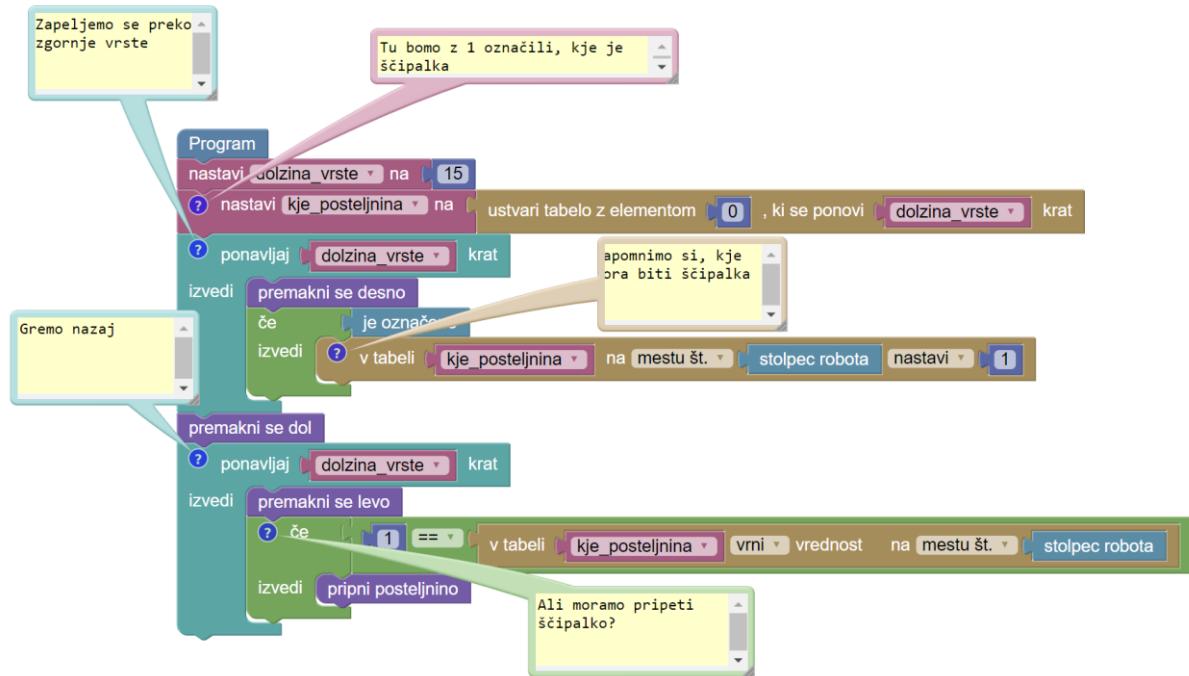
Piškove volitve



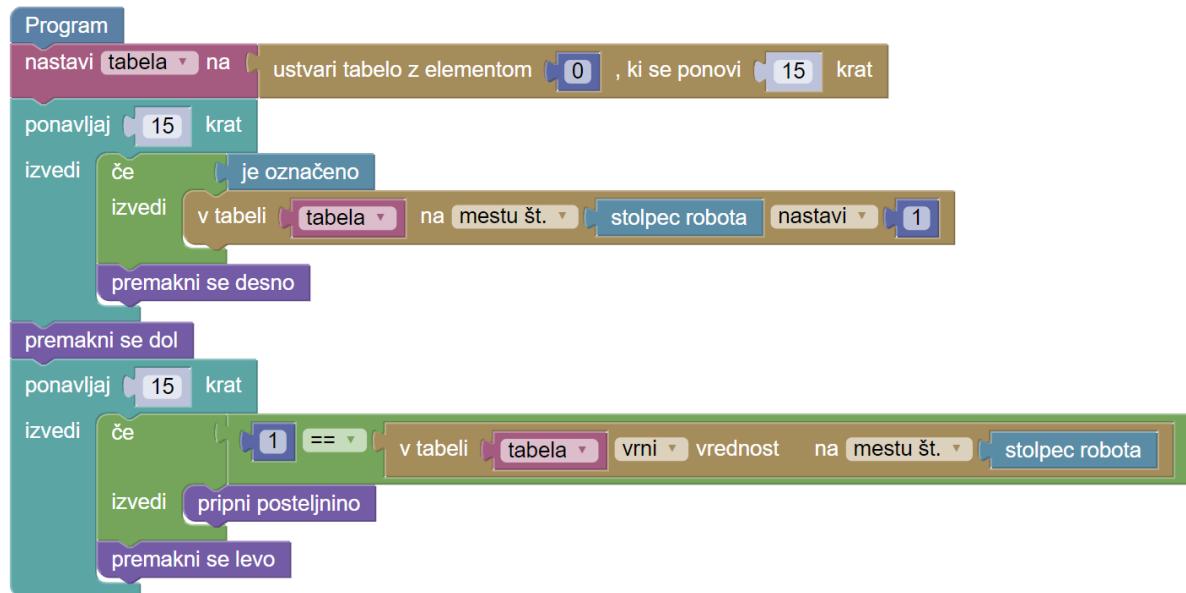
Pišek na orientacijskem tekmovanju



Obešanje posteljnine

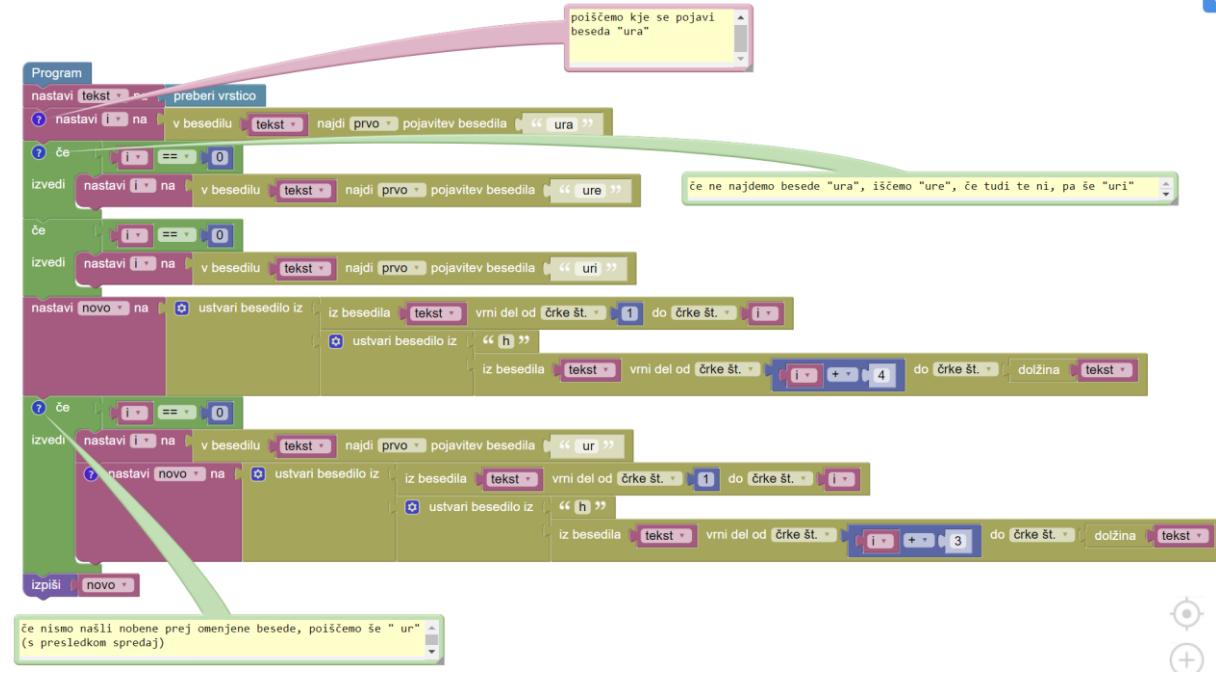


V drugi različici smo le zamenjali vrstni red primerjanja in premikanja. Prav tako se nismo "potrudili" z imeni spremenljivk.

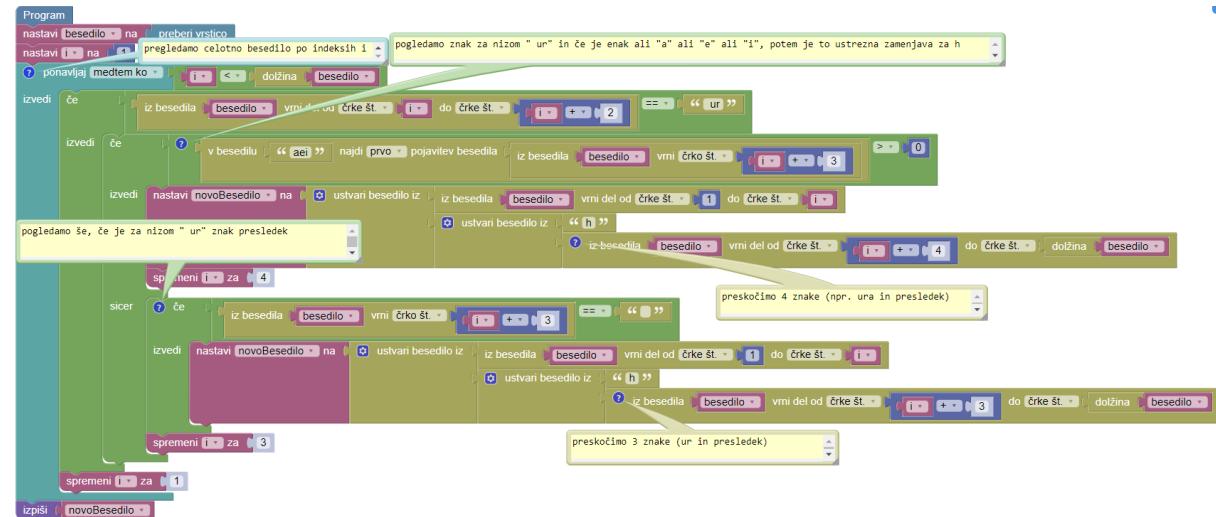


Zapis ure

1. možnost:



2. možnost:



3. možnost

```

Program
nastavi besedilo na preberi vrstico
nastavi novo na " "
za i od 1 do dolžina besedilo s korakom 1
izvedi če "ur" == iz besedila besedilo vrn del od črke št. i do črke št. i + 2
izvedi nastavi novo na ustvari besedilo iz novo
če v besedilu "aei" najdi prvo pojavitev besedila iz besedila besedilo vrn črko št. i + 3 != 0
izvedi spremeni i za 1
sicer nastavi novo na ustvari besedilo iz novo
iz besedila besedilo vrn črko št. i
izpiši novo

```

Kozmonavt

```

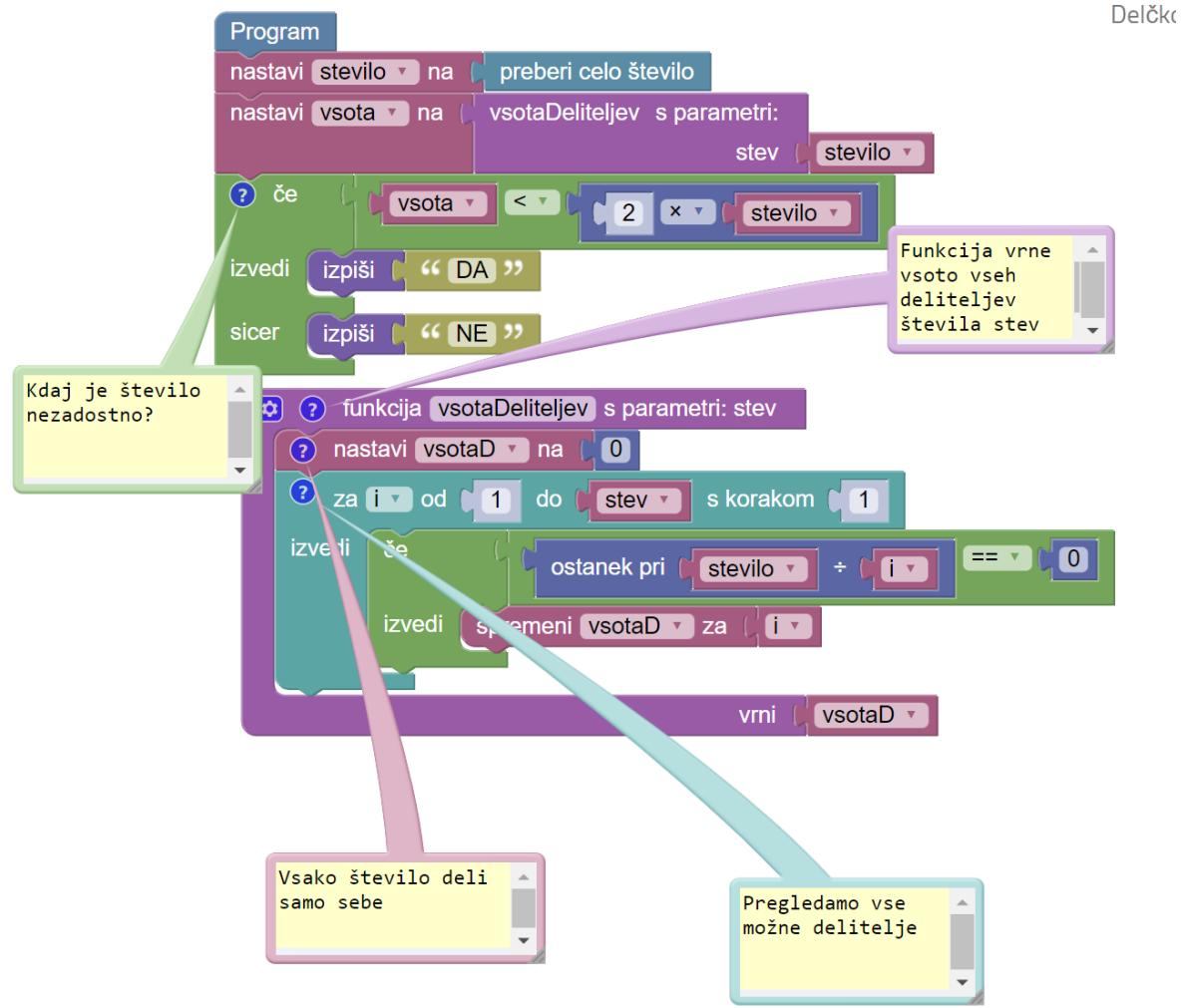
Program
premakni se naprej
nastavi okvare na število okvar
? ponavljam okvare krat
izvedi ponavljam dokler ni je okvara
izvedi če je pot == 1
izvedi obrni se levo
premakni se naprej
sicer če je pot spredaj
izvedi premakni se naprej
sicer obrni se desno
popravi okvaro

```

Three yellow message boxes are shown:

- Poiskati moramo vse okvare**
- Vedno najprej posk simo levo**
- Näsllednje možnost je pot naprej**

Nezadostna števila



Pišek gleda tekmo

Rešitev bomo tokrat prikazali v dveh delih, ker je zadnji izraz zelo širok.

Program

```
when green flag clicked
    [set [dosezeni_kosi v-]
    [set [rezultat v-]
    [repeat (preberi celo število)
        [set [dosezeni_kosi v-]
        [set [rezultat rezultat + dosezeni_kosi]
        [if (dosezeni_kosi = 0) then
            [set [rezultat rezultat + 1]
        end
        [if (dosezeni_kosi = 1) then
            [set [rezultat rezultat + 1]
        end
        [if (dosezeni_kosi = 2) then
            [set [rezultat rezultat + 2]
        end
        [end
    end
    say (Tekma je lahko potekala na toliko načinov:) (3)
    end
    end
    end
```

funkcija kosi s parametri: tocke

```
function [kosisparametri: tocke]
    if (tocke = 0) then
        vrni 1
    end
    if (tocke = 1) then
        vrni 1
    end
    if (tocke = 2) then
        vrni 2
    end
    end
end
```

preberi celo število

```
when green flag clicked
    [set [dosezeni_kosi v-]
    [set [rezultat v-]
    [repeat (preberi celo število)
        [set [dosezeni_kosi v-]
        [set [rezultat rezultat + dosezeni_kosi]
        [end
    end
    say (Tekma je lahko potekala na toliko načinov:) (3)
    end
    end
    end
```

vrni

```
when green flag clicked
    [set [dosezeni_kosi v-]
    [set [rezultat v-]
    [repeat (preberi celo število)
        [set [dosezeni_kosi v-]
        [set [rezultat rezultat + dosezeni_kosi]
        [end
    end
    say (Tekma je lahko potekala na toliko načinov:) (3)
    end
    end
    end
```

Bolj pregledna rešitev bi bila z uvedbo spremenljivk, ki vrnejo koliko košev je padlo, če je bil zadnji doseženi met za 1, 2 ali za 3 točke. Potem bi napisali

Program

```
nastavi dosezeni_kosi na preberi celo število
nastavi rezultat na kosi s parametri:
    tocke dosezeni_kosi
nastavi besedilo na "Tekma je lahko potekala na toliko načinov: "
k besedilo dodaj besedilo rezultat
izpiši besedilo
```

funkcija kosi s parametri: tocke

```
če tocke == 0 vrni 1
če tocke == 1 vrni 1
če tocke == 2 vrni 2
nastavi kosi_1 na kosi s parametri:
    tocke tocke - 1
nastavi kosi_2 na kosi s parametri:
    tocke tocke - 2
nastavi kosi_3 na kosi s parametri:
    tocke tocke - 3
vrni kosi_1 + kosi_2 + kosi_3
```

A žal ta rešitev ni pravilna. Spremenljivke v Blocklyu niso "lokalne", torej ne dobimo svoje "sveže" spremenljivke kosi_1 ob vsakem klicu. Zato moramo žal napisati tisto "kačo" v rešitvi.