

NEOBVEZNI IZBIRNI PREDMET RAČUNALNIŠTVO (petkova skupina) – 10. 4. 2020

Upam, da ste se s Piškom pozabavali in vam je uspelo rešiti čim več nalog. Letos se žal zaradi izrednih razmer v katerih smo se znašli, žal ne bomo mogli udeležiti tekmovanja v programiranju, saj ga ne bodo izpeljali. Upam, da bo to možno že naslednje leto, saj menim, da je med vami kar nekaj zelo uspešnih programerjev, ki bi se na tekmovanju dobro izkazali.

Vesela bom kakšne povratne informacije o tem, kakšne so se vam zdele naloge, katere so vam bile najbolj zanimive in katere najtežje.

Zdaj pa pojdimo k novi zadevi. Ker je Python mogoče za delo na daljavo malo težje usvojiti, bomo poskusili vsaj malo približati ta programski jezik s pomočjo Blockly-a. To je jezik v katerem ste že ustvarjali v Pišku, v ozadju pa se pravzaprav skriva Python. Kako se bomo lotili dela?

Na voljo imamo enkratni interaktivni učbenik. Kaj to pomeni, da je interaktivni. Obenem, da v njem lahko preberemo marsikaj zanimivega, nove razlage, ipd. lahko v njega tudi rešujemo in dobimo tudi povratno informacijo.

Interaktivni učbenik uporabljamo tako, da najprej preberemo vsebino, ki je razložena. Povsod imate zraven sličice blokov, ki jih uporabljamo za sestavljanje programov in razlago, kaj posamezen blok pomeni. Potem sledi vaja. Sami boste delovanje Blockly-a tudi preizkusili.

Začeli bomo na dani povezavi: <https://lusy.fri.uni-lj.si/ucbenik/prog/1221/index2.html>

Najprej boste prebrali nekaj teorije o Blockly-u - kaj je tolmač in kaj prevajalnik. Kot vidite spodaj na sliki je to stran 4 (če želite, si lahko preberete vse strani od 1 do 4).

Programiranje si algoritmi

Slikovno programiranje

Uvod

Osnovni koncepti programiranja

Izboljšava samostojnih programov in pogosti stavki

Zanke

Tabele



Funkcije

Nizi

Urejevalnik

Interaktivni tolmač

V interaktivnem učbeniku bomo uporabljali interaktivno okolje Blockly za slikovno programiranje, ki nam bo omogočalo, da bomo lahko vse primere preizkusili kar neposredno v samem učbeniku. Poleg tega to interaktivno okolje omogoča tudi prikaz koda v besedilnem programskem jeziku python. Tako bomo lahko primerjali kodo vsakega primera v slikovnem programskem jeziku, s kodo v besedilnem programskem jeziku.

Slikovni programski jezik

Slikovni ali grafični programski jezik (ang. *Visual programming language* - VPL) je vsak programski jezik (ali okolje), ki uporabniku omogoča ustvarjanje programov tako, da uporabnik premika ali razporeja elemente slikovnega programskega jezika. Prednost slikovnih programskih jezikov je v tem, da omogočajo ustvarjanje ali gradnjo sintaktično pravih programov, saj lahko uporabnik združuje elemente na točno določen način. S tem se izogne sintaktičnim napakam, kar posledično olajša učenje programskega jezika začetnikom. Ti se lahko namesto s sintakso, t.j. slovnico pravilnosti programa, ukvarjajo s postopkom, ki bo privedel do rešitve problema.

Obstaja cela množica slikovnih programskih jezikov, ki jih lahko uporabljamo za različne namene. Najbolj znani slikovni programski jeziki za učenje programiranja so: Alice, App Inventor, Blockly, Flowgorithm, Greenfoot, Lego Mindstorms NXT, Raptor, Scratch, Snap! in številni drugi.

V tem interaktivnem učbeniku bomo uporabljali interaktivno okolje Blockly v kombinaciji s besedilnim programskim jezikom python. Blockly namreč omogoča, da uporabnik lahko gradi kodo programskega jezika tako, da razporeja in združuje elemente slikovnega programskega jezika. Hkrati pa se ta koda samodejno pretvaja v izvorno kodo v besedilnem programskem jeziku.

Razlika med tolmačem in prevajalnikom

Večina besedilnih programskih jezikov je nekakšna zmes preproste angleščine in matematike, kar programeru omogoča, da se osredotoči na problem, ki ga želi rešiti, ne pa na strojno opremo računalnika, na katerem dela. Računalnik pa razume zgolj **strojni jezik**, ki vsebuje ukaze, kot je npr. "Prenesi podatek iz pomnilne celice A v pomnilno celico B." Zato se mora naš program pred ali med izvajanjem pretvoriti v enakovreden program v strojnem jeziku.

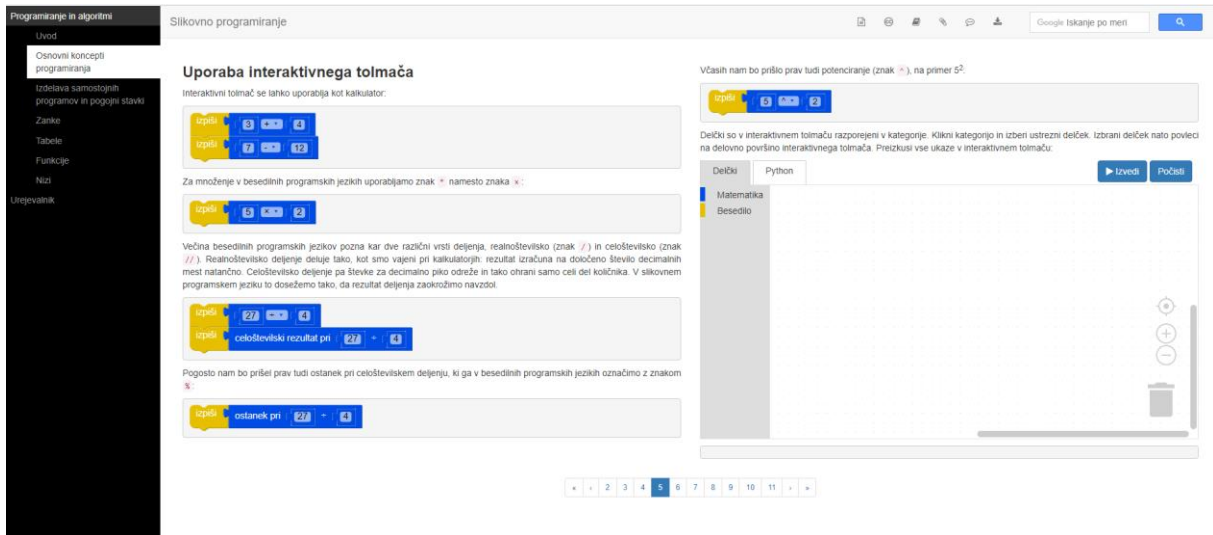
Ce se pretvorba v celoti izvrši pred izvajanjem, govorimo o **prevajanju**; poseben program, ki mu pravimo **prevajalnik**, prevede naš program v enakovreden program v strojnem jeziku.

Druga možnost pa je **tolmačenje** program po imenu **tolmač** bere naš program ukaz za ukazom in računalniku sproti podaja enakovredne ukaze v strojnem jeziku. Tolmačeje je počasnejše od izvajanja prevedenih programov, vendar pa se jeziku, ki se tolmačijo, praviloma enostavnejši od tistih, ki se prevajajo.

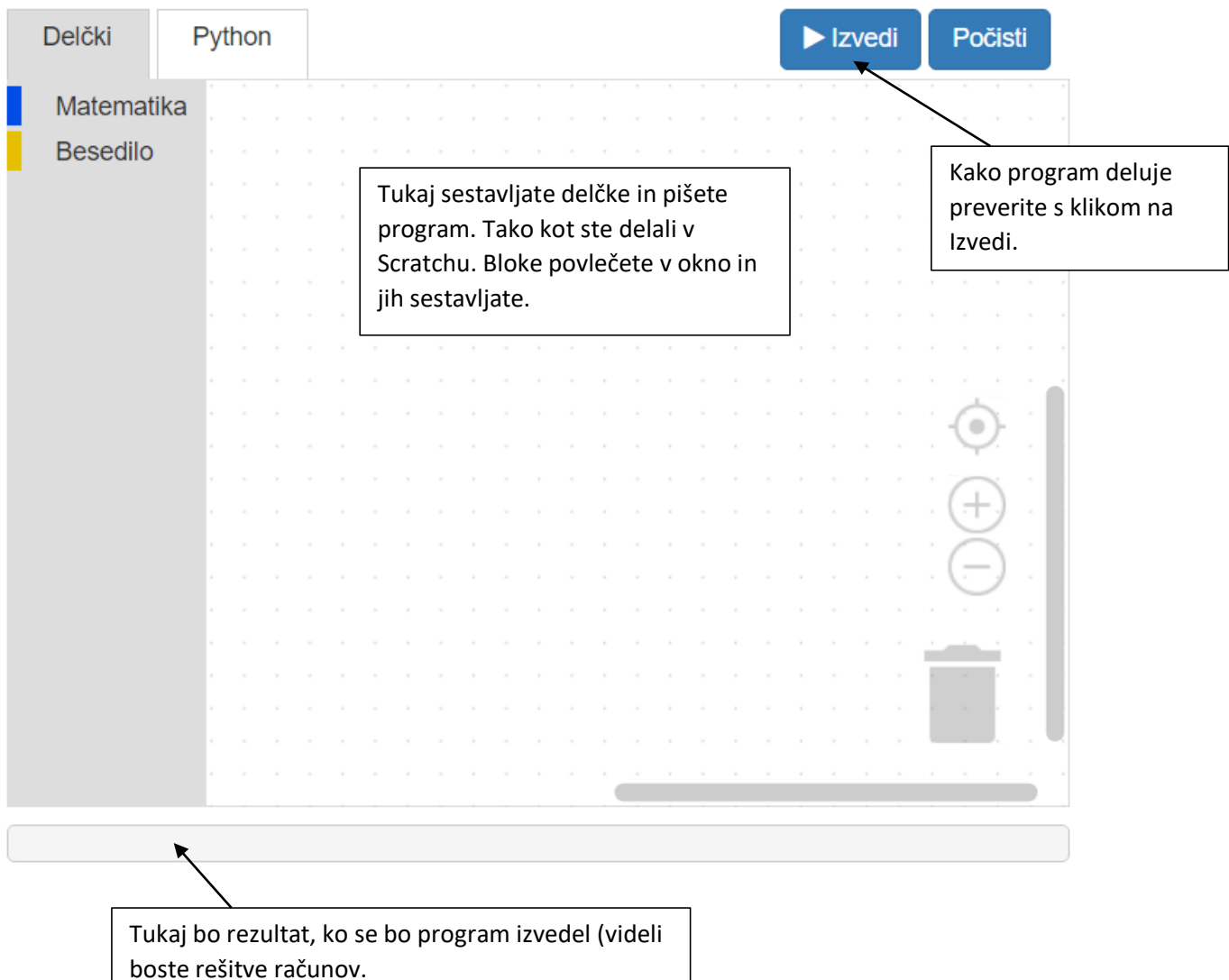
Nekateri programski jeziki – med njimi tudi python – uporabljajo kombinacijo obeh pristopov: program se najprej v celoti prevede v nekakšen vmesni jezik, dobitni prevod pa se nato tolmači.

1 2 3 4 5 6 7 8 9 10 11

Na strani 5 boste ugotovili kako uporabljamo interaktivni tolmač (zelo podobna stvar, kot tista, ko smo v šoli s Pythonom spoznavali matematične funkcije in ste Python uporabljali kot kalkulator).



Preglejte vse primere, preberite kako spreminjate računsko operacijo in opazujte, kako so bloki med seboj sestavljeni. Ko boste primere »naštudirali« imate na voljo delovni prostor za testiranje funkcij. Tukaj lahko preizkusite, kako Blockly računa z vsemi operacijami, ki so na voljo.



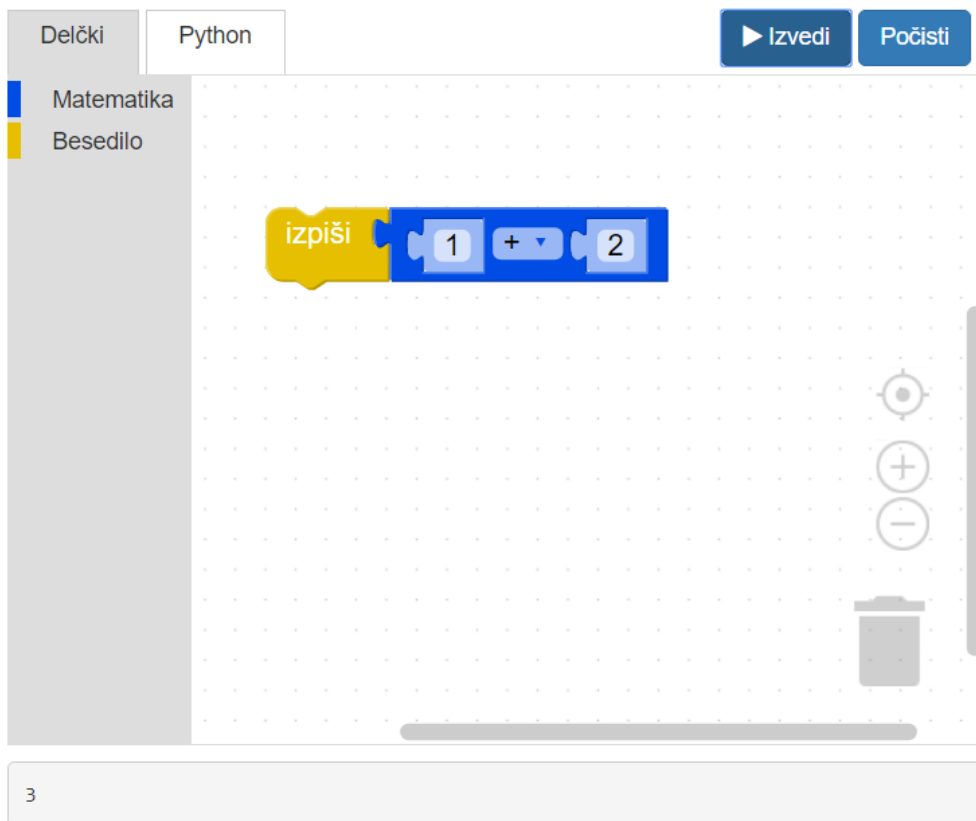
Kako dobite delčke? Kliknete na **Matematika** in odprejo se vsi delčki, ki so na voljo za matematične operacije, kliknete na **Besedilo** in na voljo imate delček izpiši (ukaz, ki izpiše rezultat).

The image shows the Scratch interface with the 'Delčki' (Blocks) palette open to the 'Matematika' (Math) category. The 'Python' tab is also visible. At the top right, there are buttons for 'Izvedi' (Run) and 'Počisti' (Clear). The workspace contains several mathematical blocks: a '0' block, a '1 + 1 + 2' block, an 'ostanek pri 64 ÷ 10' (remainder) block, and a 'celoštevilski rezultat pri 64 ÷ 10' (integer result) block. On the right side, there are navigation icons for zooming in (+), zooming out (-), and a trash can icon.

The image shows the Scratch interface with the 'Delčki' (Blocks) palette open to the 'Besedilo' (Text) category. The 'Python' tab is also visible. The workspace contains two text blocks: an 'izpiši " abc "' (print) block and an 'izpiši 1 + 2' (print) block. The 'Matematika' category is also visible in the palette.

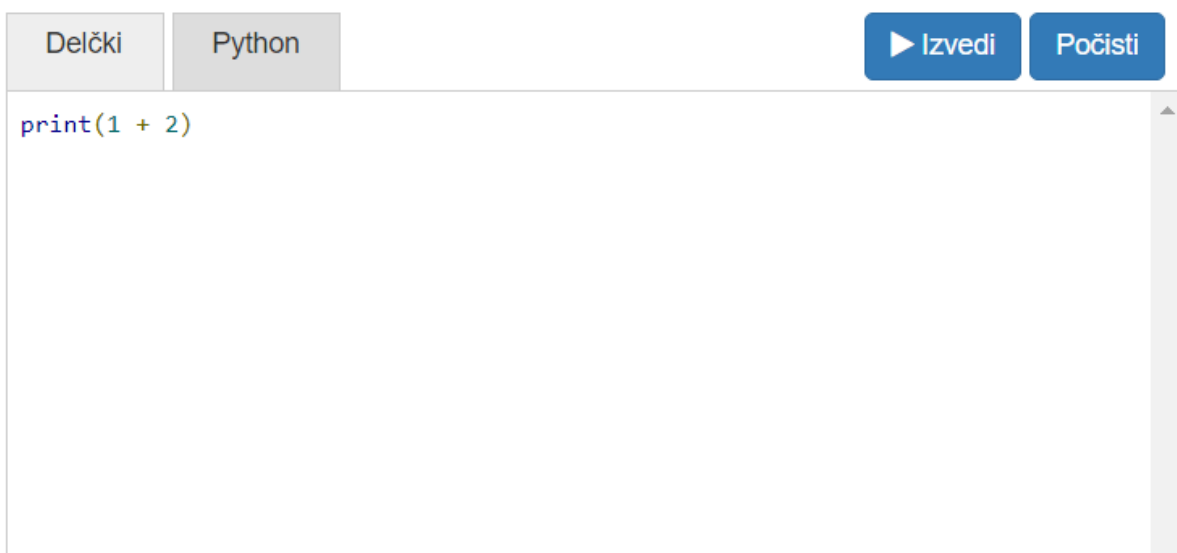
Primer:

Sestavila sem program, ki izračuna vsoto dveh števil. V okno za sestavljanje programa sem sestavila ustrezne delčke, spodaj v sivem oknu pa se izpiše rezultat 3.



The image shows the Blockly interface. At the top, there are two tabs: 'Delčki' (selected) and 'Python'. To the right are two buttons: 'Izvedi' (Execute) and 'Počisti' (Clear). On the left, there is a category list with 'Matematika' (Math) and 'Besedilo' (Text). The main workspace contains a script with a yellow 'izpiši' (print) block containing a blue '1 + 2' block. Below the workspace is a grey output window displaying the number '3'. On the right side of the workspace, there are icons for zooming in (+), zooming out (-), and a trash can.

Če preklopim na zavihek **Python**, si lahko ogledate, kako bi to kodo zapisali v Pythonu, vendar v tem zavihku vi ne morete programirati. Pravzaprav se na nek način učimo tudi Python, se strinjate? Da dobite nazaj Blockly kliknete na **Delčki**.



The image shows the Python interface. At the top, there are two tabs: 'Delčki' and 'Python' (selected). To the right are two buttons: 'Izvedi' (Execute) and 'Počisti' (Clear). The main workspace contains a text area with the code `print(1 + 2)`. On the right side of the workspace, there is a vertical scrollbar.

Vaša naloga je torej naslednja:

Preberete strani, ki sem jih napisala, pregledate primere, kako interaktivni tolmač uporabimo kot kalkulator. Nato boste sami preizkusili vse delčke, ki jih lahko najdete v urejevalnem oknu (vse možne računske operacije). Bloke dodajate enega pod drugim, ko boste program izvedli, bodo rezultati v takšnem vrstnem redu, kot ste v delovni površini sestavljali bloke. Da bom videla, da ste to uspešno sestavili, mi boste, ko boste z delom končali naredili posnetek zaslona in poslali na e-poštni naslov (nika.petelin@sola.velike-lasce.si). Posnetek zaslona (print screen) naredite s tipko, ki jo najdete na tipkovnici (PrtSc, PrintScreen). Ko boste tipko pritisnili, lahko sliko zaslona nalepite kar v Word ali Slikar (lahko tudi direktno v novo e-sporočilo). V Word ali Slikar sliko nalepite z desnim klikom na miško in ukazom Prilepi. Nato dokument shranite in ga pošljite po e-pošti kot prilogo. Vaše »potrdilo« o opravljeni nalogi, mi pošljite do naslednjega **četrтка, 16. 4. 2020.**