Postov problem in Turingov stroj[1]

Rok Gregorič, Vesna Iršič, Anja Petković, David Gajser (mentor)

→ So problemi v matematiki, tako kot v življenju, ki jih preprosto (še) ne znamo rešiti. So pa problemi, ki jih z nobenim končnim postopkom (t. j. algoritmom) niti ne moremo rešiti. Kaj pomeni, da z algoritmom rešimo problem in kakšen bi bil primer problema, kjer to ni mogoče?

Postov problem

Na voljo imamo domine

01	1	010	00
0101	0	1	0

ki jih želimo zložiti v vrsto eno za drugo tako, da bomo zgoraj in spodaj dobili enak niz znakov. Pri tem lahko vsako domino uporabimo poljubno mnogokrat, vendar smemo uporabiti le končno mnogo domin.

Ta problem ni preveč težak in ga rešimo, recimo, tako, da dane domine zložimo v vrsto[2]

00	1	010	1	010	01	00	1	010	1	01	01	01
0	0	1	0	1	0101	0	0	1	0	0101	0101	0101

Enak problem pri danih dominah

100	0	1
1	100	0

je mnogo težje rešiti, saj njegovo (najkrajšo) rešitev sestavlja 75 domin, pri čemer obstajata dve različni rešitvi te dolžine.

Problem lahko zastavimo za poljubno število domin:

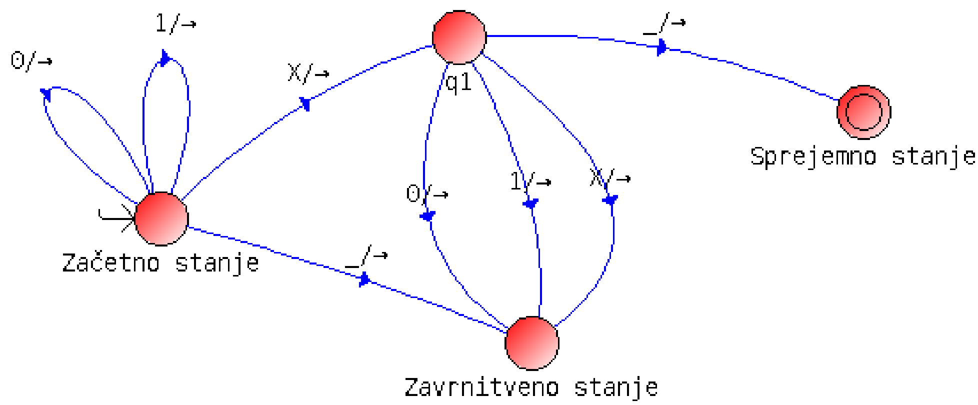
Postov problem. Danih imamo končno mnogo domin, na zgornjem in spodnjem delu vsake domine pa je zapisan niz znakov. Ali lahko te domine zložimo v končno vrsto eno za drugo tako, da bomo zgoraj in spodaj dobili enak niz znakov? Pri tem lahko vsako domino uporabimo poljubno mnogokrat.

Ta problem je prvi zastavil ameriški matematik Emil Post leta 1946 in je zanimiv med drugim tudi zato, ker se izkaže, da ga ni mogoče rešiti z računalnikom. Ni ga mogoče rešiti z računalnikom?

SLIKA 1.
Avtorji (iz leve): David, Anja, Vesna, Rok.

[1] Članek je nastal na poletnem taboru MaRS 2013 (Matematično Raziskovalno Srečanje za srednješolce).

[2] Obstaja tudi krajša rešitev, ki vsebuje le štiri domine. Jo najdeš?



SLIKA 2.
Grafični prikaz Turingovega stroja M. Iz vsakega stanja, ki ni sprejemno ali zavrnitveno, gredo natanko štiri puščice – za vsak znak iz Γ ena. Znaki → na puščicah nam nakazujejo, da se glava stroja zmeraj premika v desno. Ko stroj v stanju A prebere znak a, glava na trak napiše a, se premakne v desno, stroj pa preide v stanje, v katerega kaže puščica iz A z znakom a. Če bi imeli tak stroj, da glava znakov na traku ne bi ohranjala, bi morali na vsako puščico dodati še nov znak.

Odločljivi in neodločljivi problemi

Problemom, na katere lahko odgovorimo samo z *da* ali *ne*, pravimo *odločitveni problemi*. Ker nas zanima predvsem Postov problem, ki je odločitveni, se bomo od tu naprej ukvarjali le s takšnimi problemi. Pravimo, da je odločitveni problem *odločljiv*, če ga lahko rešimo z algoritmom, t. j. če obstaja algoritem, ki nam pove, kdaj je pravilen odgovor *da* in kdaj *ne*.

Poglejmo si preprost primer odločitvenega problema.

Ime: PALINDROM
Vhod: Niz ničel in enic
Vprašanje: Ali se niz iz leve proti desni prebere enako kot iz desne proti levi?

Primeri vhodov, za katere je odgovor *da*, so npr. nizi 110011, 100001, 101010101 …

Problem PALINDROM znamo rešiti z algoritmom; verjetno je vsak nadobuden bralec že ugotovil, kako. Začnemo lahko npr. s primerjavo skrajnega levega in skrajnega desnega znaka besede. Če nista enaka, je odgovor *ne*. Če pa sta, ju lahko izbrišemo in nadaljujemo na krajši besedi, vse dokler nam ne ostane le še en znak ali pa znakov zmanjka. V obeh primerih je odgovor *da*.

Torej je PALINDROM odločljiv problem. Zanimivo pa je, da obstajajo tudi problemi, ki niso odločljivi. Takšnim pravimo *neodločljivi problemi*.

Church-Turingova teza

Ključno vlogo v definiciji odločljivosti problema ima *algoritem*. Problem je namreč odločljiv natanko tedaj, ko obstaja algoritem, ki ga reši. Neformalno algoritem razumemo kot končno zaporedje preprostih ukazov, ki jih moramo izvesti, da pridemo do rezultata, npr. računalniki probleme rešujejo z algoritmi. Ali lahko pomen besede *algoritem* tudi bolj formalno opredelimo?

Matematiki so se na več načinov trudili odgovoriti na to vprašanje, na koncu pa se je izkazalo, da je bilo veliko teh načinov povsem enako dobrih. Ena izmed opredelitev pojma algoritem je s pomočjo Turingovega stroja,[3] tj. naprave, ki jo bomo podrobneje opisali v naslednjem razdelku.

Church-Turingova teza. Problem lahko rešimo z algoritmom natanko tedaj, ko obstaja Turingov stroj, ki reši ta problem.

Church-Turingova teza je v rabi od leta 1936 in je v teoretičnem računalništvu splošno sprejeta. Pove nam, da je Turingov stroj enako dober kot katerikoli drug model za algoritem. Če torej najdemo algoritem za reševanje nekega problema, lahko isti problem rešimo tudi s Turingovim strojem. Zaradi teze

[3] Preostali dobro poznani opredelitvi algoritma sta s pomočjo lambda računa in s pomočjo rekurzivnih funkcij.

se torej ni potrebno spuščati v podrobnosti, ki jih zahteva delo s Turingovim strojem, ampak lahko delamo na višjem nivoju abstrakcije.

Turingov stroj

Leta 1936 je angleški matematik Alan Turing zasnoval *Turingov stroj*, ki je preprost teoretični model računalnika.[4] Sestavljajo ga v eno smer neskončen trak, glava in »program«, ki pove pravila, kako naj se glava premika po traku levo in desno ter ga spreminja. Po vsakem premiku glava prebere znak, zapisan pod njo na traku in ga prepiše z drugim znakom, lahko tudi enakim. Pri tem stroj prehaja preko različnih stanj, dokler ne pride do sprejemnega ali zavrnitvenega stanja – takrat se ustavi. Lahko pa se zgodi tudi, da stroj nikoli ne pride v eno od teh dveh stanj in se nikdar ne ustavi.

Turingov stroj je natančno določen s:

- končno množico stanj Q, v kateri so tudi paroma različna stanja q_0, q_s in q_z, ki jim pravimo *začetno, sprejemno* in *zavrnitveno* stanje,
- končno množico Γ, ki vsebuje znake, ki jih Turingov stroj lahko uporabi. Ta množica vsebuje tudi znake, s katerimi stroju podamo vhod, ter poseben znak, ki nikoli ni del vhoda: *prazen znak*,
- prehodno funkcijo (bistvo »programa«) $\delta : Q\times\Gamma \to Q\times\Gamma\times\{L, D\}$, kjer L pomeni premik v levo, D pa v desno.

Razložimo pomen prehodne funkcije δ, ki je ključen del Turingovega stroja. Denimo, da je stroj v stanju q in je pod glavo na traku zapisan znak a. Če je $\delta(q, a) = (r, b, L)$, bo stroj izbrisal znak a in na njegovo mesto zapisal b, pri tem bo prešel v stanje r ter se premaknil v levo. Če bi bila zadnja komponenta D, bi se stroj premaknil v desno. Vhodne podatke, oz. vhod Turingovemu stroju podamo kot niz znakov, ki se nahaja na začetku traku, preostanek traku pa je prazen, t. j. zapolnjen s praznimi znaki. Glava se na začetku nahaja na najbolj levem delu traku, torej na prvem znaku vhoda. Stroj začne v začetnem stanju in deluje tako, kot mu predpisuje prehodna funkcija δ. Takoj, ko preide v sprejemno ali zavrnitveno stanje, se ustavi.

Za boljšo predstavo bomo opisali zelo preprost Turingov stroj M, ki preveri, ali je vhod sestavljen le iz znakov 0 in 1 ter se zaključi z znakom X. Naj bo $\Gamma = \{0, 1, X, \_\}$ množica znakov, ki jih M lahko uporabi (_ označuje prazen znak). Prehodna funkcija deluje na sledeč način (glej tudi sliko 2):

- Če je stroj v začetnem stanju in glava prebere 0, potem glava zapiše 0, se premakne v desno in stroj ostane v začetnem stanju.
- Če je stroj v začetnem stanju in glava prebere 1, potem glava zapiše 1, se premakne v desno in stroj ostane v začetnem stanju.
- Če je stroj v začetnem stanju in glava prebere X, potem glava zapiše X, se premakne v desno in stroj preide v stanje q_1.
- Če je stroj v stanju q_1 in glava prebere prazen znak, stroj sprejme vhod, torej preide v sprejemno stanje.
- Če se zgodi karkoli razen zgornjega (npr. stroj je v začetnem stanju in glava prebere prazen znak), stroj zavrne vhod, torej preide v zavrnitveno stanje.

Izkaže se, da kljub preprosti definiciji Turingovega stroja, ne moremo preprosto ugotoviti, na katerih vhodih se Turingov stroj ustavi.

Zaustavitveni problem. Ali se dani Turingov stroj M ustavi na danem nizu iz ničel in enic?

Ta problem je eden najpomembnejših in najbolj znanih neodločljivih problemov. Njegova neodločljivost je bila dokazana že v tridesetih letih prejšnjega stoletja, a bomo zaradi poljudnosti članka dokaz izpustili.

Zaključek

Na začetku smo trdili, da Postovega problema ne moremo rešiti z računalnikom. Kako bi sploh lahko to utemeljili?

Ker lahko s Turingovim strojem rešimo vse, kar lahko rešimo tudi z računalnikom, je dovolj pokazati, da Postovega problema ni moč rešiti s Turingovim strojem. Izkaže se, da bi v primeru, da bi nek

[4] Izkaže se, da lahko Turingov stroj zaradi neskončnega traku (v teoriji) reši precej več problemov kot katerikoli računalnik na svetu (ki je seveda končen). Če pa bi računalnik imel na voljo neskončno trdega diska, bi Turingov stroj rešil natanko tiste probleme kot računalnik.

Turingov stroj rešil Postov problem, lahko skonstruirali algoritem, ki bi rešili tudi zaustavitveni problem, kar pa ni mogoče. Dokaz lahko bralec najde v [1, pogl. 5.2].

Postov problem torej ni odločljiv in tako ni smiselno iskati algoritma, ki bi ga reševal. To pa še ne pomeni, da se s Postovim problemom ni vredno ukvarjati.

Lahko se vprašamo, ali je odgovor Postovega problema pri konkretnih naborih domin *da* ali *ne*. Izmed problemov s tremi dominami, kjer je največja dolžina niza znakov na dominah tri, so razrešeni že skoraj vsi primeri. To pomeni, da je za vsak primer znana ustrezna postavitev domin v vrsto, ali pa je dokazano, da taka postavitev ne obstaja. Zadnji odprt primer tega tipa je podan z dominami

10	0	001
0	001	1

Ko boste imeli trenutek prostega časa, se lahko z njim pozabavate tudi vi.

Literatura

[1] M. Sipser, *Introduction to the Theory of Computation, Second Edition*. Course Tehnology, 2006.

[2] L. Zhao, *PCP: a Nice Problem*, `http://webdocs.cs.ualberta.ca/~games/PCP/`, citirano dne 21. 8. 2013.

×××

Naloga

↓↓↓

MARKO RAZPET

→ **Izračunaj** A_{10}**,** A_{100} **in** A_{1000}**, če veš, da je** $A_1 = 4$ **in**

- $$A_{n+1} = \frac{A_n}{1 + n^3 A_n} \qquad (n = 1, 2, 3, \ldots).$$

×××

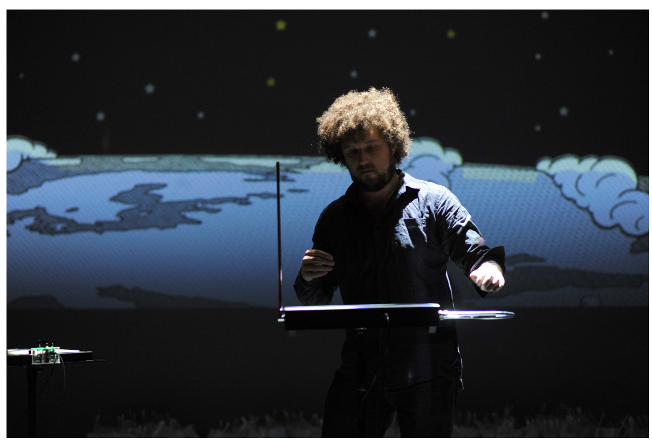
Bistroumi 2014

SREČANJE MLADIH MATEMATIKOV, FIZIKOV IN ASTRONOMOV

↓↓↓

BOŠTJAN KUZMAN, FOTO: JAN ŠUNTAJS

→ **V letošnjem letu se je tekmovanj iz matematike, fizike, astronomije, razvedrilne matematike in poslovne matematike za različne stopnje osnovne in srednje šole v organizaciji DMFA Slovenije udeležilo več kot 125.000 učencev in dijakov, podeljenih pa je bilo skupaj 819 zlatih priznanj (`http://www.dmfa.si/Aktualno/Statistika.html`). Med prejemniki zlatih priznanj je bilo 171 nagrajencev skupaj z družinskimi člani, mentorji in predstavniki šol povabljenih na tradicionalno podelitev nagrad, ki je pod naslovom Bistroumi 2014 potekala v soboto, 24. maja, v Linhartovi dvorani Cankarjevega doma v Ljubljani.**



SLIKA 1.
Glasbenik in fizik Janez Dovč igra na theremin.