ACTA MATHEMATICA SPALATENSIA
Series didactica
Vol.4 (2021) 113–124
Zaprimljen: 12. srpnja 2020.
Prihvaćen: 22. kolovoza 2020.

Bealova jednadžba i veliki brojevi – potraga u programskom jeziku C++

Josip Ćatić, Davor Damjanović, Filip Đuričković, Leon Gal, Zvonimir Haramustek, Leon Kodžoman, Josip Koprivnjak, Borna Krušlin, David Majdandžić, Neven Marić, Luka Miler, Tomislav Rudec, Krešimir Turkalj, Ivan Vazler

Sažetak

Postoje li prirodni brojevi A, B i C, x, y i z za koje je $A^x + B^y = C^z$, a za koje vrijedi da su A, B i C relativno prosti, a x, y i z su veći ili jednaki 3 (Bealova pretpostavka, [1]) jedno je od najpopularnijih pitanja današnje matematike. U ovom članku doprinijet ćemo tvrdnji da takvih brojeva nema tako što ćemo pokazati da jednadžba $A^x + B^y = C^z + p$ već u prvih milijun brojeva ima rješenja za sve cijele brojeve p iz intervala $[-10, 10]$, osim za $p = 0$. Slične su ovoj jednadžbi Pitagorina i Fermatova jednadžba, pa ćemo originalnim paralelnim programom u programskom jeziku C++ istražiti koliki je odmak slučajno izabrane trojke prirodnih brojeva od uvjeta Bealove, Fermatove i Pitagorine jednadžbe u prvih milijun, milijardu ili bilijun prirodnih brojeva.

U točki 1 i 2 dajemo matematičke osnove područja, u sljedeće tri točke računalnim programom analiziramo rješenja triju navedenih jednadžbi, dok je zadnja točka zaključak.

Ključni pojmovi: Fermat, Beal, Pitagora, jednadžbe, slutnje, induktivni zaključak.

1. Uvod

Pitagorine trojke su trojke (a, b, c) prirodnih brojeva koje zadovoljavaju Pitagorinu jednadžbu

$$a^2 + b^2 = c^2.$$

Znamo da je takvih trojki beskonačno puno i da su neka od rješenja (sve „primitivne trojke”, tj. one kod kojih su a, b i c relativno prosti) dana izrazima:

$$a = m^2 - n^2,\ b = 2mn,\ c = m^2 + n^2, \text{te}$$
$$a = 2mn,\ b = m^2 - n^2,\ c = m^2 + n^2,$$

gdje su m i n parametri, proizvoljni relativno prosti prirodni brojevi različite parnosti i $m > n$. Dokaz ove tvrdnje i detaljan uvod u problematiku dan je u [2]. (Rješenja su uređene trojke, pa trojke $(5, 12, 13)$ i $(12, 5, 13)$ smatramo različitim rješenjima.)

Time smatramo tu jednadžbu do kraja riješenom, tj. tu više nema nikakvih rasprava o postojanju ili obliku rješenja.

Veliki Fermatov[1] teorem [3] tvrdi da za prirodni broj n veći od 2 ne postoje trojke (a, b, c) prirodnih brojeva koje zadovoljavaju jednadžbu

$$a^n + b^n = c^n.$$

Tu je tvrdnju nakon više od 350 godina od njezina postavljanja dokazao 1994. godine engleski matematičar Andrew Wiles.

2. Bealova pretpostavka

Za razliku od gornja dva slučaja, ne zna se postoje li prirodni brojevi A, B i C koji su relativno prosti (najveći prirodni broj s kojim su A, B i C djeljivi je 1) te x, y i z svi veći ili jednaki 3 i za koje je

$$A^x + B^y = C^z.$$

Tvrdnja da takvi brojevi ne postoje zove se Bealova pretpostavka, a gornju ćemo jednadžbu zvati Bealova jednadžba.

Ako zanemarimo uvjet: A, B i C moraju biti relativno prosti, dobivamo beskonačno puno rješenja gore navedene jednadžbe. Npr.

$2^x + 2^x = 2^{x+1}$ za bilo koji prirodni broj x (A, B i C su djeljivi s 2 pa nisu relativno prosti),

$3^3 + 6^3 = 3^5$, (A, B i C su djeljivi s 3 pa nisu relativno prosti),

$7^6 + 7^7 = 98^3$, (A, B i C su djeljivi sa 7 pa nisu relativno prosti),

[1]Pierre de Fermat, 1607. – 1665., francuski matematičar

$162^3 + 27^4 = 9^7$, (A, B i C su djeljivi s 9 pa nisu relativno prosti),
$19^4 + 38^3 = 57^3$, (A, B i C su djeljivi s 19 pa nisu relativno prosti).

Ako zanemarimo uvjet: x, y i z su veći ili jednaki 3, Bealova jednadžba također ima beskonačno puno rješenja.
Npr. $3^2 + 4^2 = 5^2$, $2^3 + 1^5 = 3^2$, $1^1 + 1^1 = 2^1$, ...

Smatramo kako Bealova pretpostavka nije lako dokaziva, kao ni lako oboriva. Prije svega, ona je poopćenje Fermatova teorema, a za Fermatov teorem znamo koliko je dugo trebalo matematičarima da ga dokažu, kao i koliko duboke matematičke tvrdnje su u tu svrhu morali koristiti. Dalje, Bealova pretpostavka se relativno jednostavno postavlja na računalu pa je velika vjerojatnost da bi netko i računalom vjerojatno već našao takvu šestorku kada bi postojala među „manjim" prirodnim brojevima. No to su također samo pretpostavke.

Glavna ideja ovog članka je sljedeća: Promatrat ćemo redom potencije prirodnih brojeva (do što većeg iznosa potencije, koliko nam god to karakteristike računala dopuste) i provjeravati vrijedi li Bealova jednadžba za trojke tih potencija, a ako ne vrijedi bilježit ćemo za koliko desna strana jednadžbe premašuje lijevu stranu, tj. koliki su promašaji, odnosno otkloni od Bealove pretpostavke. Ako su i drugi otkloni veličina od npr. -10 do 10 rijetki, ima smisla tražiti rješenja među dalekim, velikim brojevima. Činjenica da su ostali otkloni svi postignuti, a otklona nula, i samo njega, nema, dodatni je argument koji nam sugerira kako Bealova jednadžba zaista nema rješenja.

Za početak, pokušat ćemo vidjeti koliko često za slučajno odabrane brojeve od 1 do 1 000 000 vrijedi Pitagorina jednadžba $a^2 + b^2 = c^2$, te koliko su česti mali (do veličine 10) odmaci od točnog rješenja. Zatim ćemo pogledati koliki su najčešće odmaci od rješenja Fermatove jednadžbe za koju znamo kako nema rješenja, a onda ćemo pratiti i odmake od rješenja u Bealovoj jednadžbi, pa ćemo pokušati induktivno (dakle, naravno, moguće i krivo) zaključiti ima li Bealova jednadžba rješenje ili ne.

3. Program za analizu Pitagorinih trojki

Skica računalnog programa za Pitagorine trojke:

`Za svaki prirodni broj` a `od` 1 `do` 1 000 000,
  `za svaki prirodni broj` b `od` 1 `do` 1 000 000
    `za svaki prirodni broj` c `od` 1 `do` 1 000 000
      `ako je` $a^2+b^2 = c^2+p$ `povećaj broj promašaja veličine` p,
      `polje` $[p]$ `za jedan.`

Naravno, program ne uzima u obzir sve gore navedene mogućnosti za a, b i c nego na osnovi odabranog c (samo c se kreće po cijeloj petlji),

pametno odabire a koji ne smije biti puno veći od c, a onda na osnovi c i a računa u kojemu intervalu mora biti b.
Na primjer,

$9^2 + 8^2 = 12^2 + 1$, tj. $9^2 + 8^2$ je „skoro” 12^2 pa će zbog $(9, 8, 12)$ broj promašaja od jedan biti povećan za jedan.

$8^2 + 4^2 = 9^2 - 1$, tj. $8^2 + 4^2$ je malo više od 9^2 pa će zbog $(8, 4, 9)$ broj promašaja veličine -1 biti povećan za jedan.

$3^2 + 4^2 = 5^2 + 0$ pa će zbog trojke $(3, 4, 5)$ broj promašaja od nula (to je u stvari točno Pitagorina trojka) biti povećan za jedan.

Pogledajmo izvještaj računalnog programa o tome kako su raspoređeni promašaji u traženju Pitagorinih trojki.

Tablica 1. Distribucija otklona od Pitagorinih trojki u prvih milijun prirodnih brojeva

p	Broj promašaja veličine p	p	Broj promašaja veličine p
-10	316 284	1	9 445 114
-9	416 919	2	198 414
-8	883 818	3	242 108
-7	755 977	4	9 279 728
-6	407 918	5	410 978
-5	447 334	6	298 022
-4	625 071	7	333 053
-3	577 370	8	495 767
-2	353 580	9	9 211 988
-1	249 938	10	366 166
0	3 961 284		

Iz tablice uočavamo kako je velik broj promašaja veličine nula, odnosno točnih Pitagorinih trojki. Njih je skoro 4 milijuna, što je na primjer desetak puta više od promašaja veličine -2. Drugim riječima, u prvih milijun prirodnih brojeva, Pitagorinu trojku susrećemo „često”. Gledajući promašaje veličina između -10 i 10, rjeđa je samo od trojki kojima je razlika lijeve i desne strane jednaka 1, 4 ili 9 (što je razumljivo jer je npr. $7^2 + 1^2 = 7^2 + 1$, $7^2 + 2^2 = 7^2 + 4$, $7^2 + 3^2 = 7^2 + 9$).

Zanimljivo je da već u prvih 100 prirodnih brojeva imamo točno 104 Pitagorine trojke: $(3, 4, 5)$, $(4, 3, 5)$, $(6, 8, 10)$, $(8, 6, 10)$, $(5, 12, 13)$, ... , $(96, 28, 100)$. Dakle, svaki, u prosjeku, prirodan broj manji ili jednak 100 je hipotenuza bar jednog pravokutnog trokuta s cjelobrojnim stranicama. No, neki prirodni brojevi nisu hipotenuze takvih trokuta, npr. 1, 2, 3, 4 ili 11, a neki su prirodni brojevi hipotenuze više Pitagorinih trojki. Tako je broj 65 hipotenuza kod čak 8 Pitagorinih trojki.

$$16^2 + 63^2 = 65^2, \qquad 25^2 + 60^2 = 65^2,$$
$$33^2 + 56^2 = 65^2, \qquad 39^2 + 52^2 = 65^2, \ldots$$

U prvih 100 prirodnih brojeva imamo samo 18 otklona od Pitagorine trojke za 2, tj. trojki za koje je $a^2 + b^2 = c^2 + 2$. To su na primjer $(3, 3, 4)$ ili $(5, 11, 12)$. Pokazuje se da je razliku 2 najteže postići i među brojevima manjima od 1 000 000 (kao što se vidi u tablici 1).

Zaključujemo da je brojeve a, b i c za koje vrijedi $a^2+b^2 = c^2$ lako naći u bilo kojem početnom komadu skupa prirodnih brojeva (prva trojka je $(3, 4, 5)$).

Na kraju, računalni program s gornjim kodom (ispitivanje svih potencija veličine do 1 000 000), premda se radi o kvadratima, tj. najveća potencija je samo 2, trajao bi danima i mjesecima bez dodatnih ideja i poboljšanja kako bismo izbjegli nepotrebne provjere. Optimizirani kod, bez uvođenja posebnih klasa velikih brojeva i bez paralelnog programiranja, i na bržim računalima izvršava se više sati.

4. Analiza rješenja Fermatove jednadžbe

Za $n = 3$, tablica otklona od točnih rješenja Fermatove jednadžbe $a^3 + b^3 = c^3$, tj. učestalost brojeva p, $-10 \leqslant p \leqslant 10$ za koje je $a^3+b^3 = c^3+p$, izgleda ovako:

Tablica 2. Distribucija otklona od „Fermatovih trojki” za $n = 3$ u prvih milijun prirodnih brojeva

p	Broj promašaja veličine p	p	Broj promašaja veličine p
-10	5	1	2 000 127
-9	2	2	0
-8	194	3	1
-7	2	4	0
-6	7	5	0
-5	0	6	2
-4	0	7	4
-3	0	8	2 000 205
-2	110	9	2
-1	138	10	2
0	0		

Uočavamo kako nema niti jedne trojke prirodnih brojeva među brojevima do milijun za koju vrijedi da je $a^3 + b^3 = c^3$, ali također nijedne trojke za koju je

$$\begin{aligned} a^3 + b^3 &= c^3 - 5, \\ a^3 + b^3 &= c^3 - 4, \\ a^3 + b^3 &= c^3 - 3, \\ a^3 + b^3 &= c^3 + 2, \\ a^3 + b^3 &= c^3 + 4, \\ a^3 + b^3 &= c^3 + 5. \end{aligned}$$

Također, i ostali se slučajevi pojavljuju puno rjeđe nego kod Pitagorine jednadžbe.

Veliki brojevi kod vrijednosti 1 i 8 rezultat su velikog broja trivijalnih trojki oblika $a^3 + 1^3 = c^3 + 1$ ($b = 1$ i $a = c$), $1^3 + b^3 = c^3 + 1$ ($a = 1$ i $b = c$), $a^3 + 2^3 = c^3 + 8$ ($b = 2$ i $a = c$), $2^3 + b^3 = c^3 + 8$ ($a = 2$ i $b = c$). Također, zanimljivo je da iako među prvih milijun brojeva nema trojki (a, b, c) za $p = -3$ i $p = 2$, one se ipak kasnije pojavljuju tj.

$$\begin{aligned} &569\,936\,821\,113\,563\,493\,509^3 + 472\,715\,493\,453\,327\,032^3 \\ &\qquad = 569\,936\,821\,221\,962\,380\,720^3 - 3, \end{aligned}$$

$$1\,214\,928^3 + 3\,480\,205^3 = 3\,528\,875^3 + 2.$$

S druge strane može se pokazati da za $p = 4$, $p = 5$, $p = -4$ i $p = -5$ jednadžba nema rješenja.

Za $n = 4$, tablica otklona četvrtih potencija od rješenja Fermatove jednadžbe sastoji se od samih nula, osim kod $p = 1$ gdje imamo samo trivijalna rješenja ($a = 1$ i $b = c$ ili $b = 1$ i $a = c$).

5. Analiza rješenja Bealove jednadžbe

Pitamo se koliko se često u prvih 10^{12} brojeva pojavljuje broj p veličine od -10 do 10 u „proširenoj" Bealovoj jednadžbi: $A^x + B^y = C^z + p$.

Tablica 3. Distribucija otklona od „Bealovih trojki"

p	Primjer:	$C^z < 10^9$	$10^9 < C^z < 2\cdot 10^9$	$2\cdot 10^9 < C^z < 3\cdot 10^9$	$3\cdot 10^9 < C^z < 4\cdot 10^9$	$4\cdot 10^9 < C^z < 5\cdot 10^9$	$C^z < 10^{12}$
−10	$1^3+2^4=3^4-10$	78	0	0	0	0	98
−9	$2^3+2^6=3^4-9$	68	0	0	0	0	90
−8	$3^3+6^4=11^3-8$	8	0	0	0	0	12
−7	$2^3+1^3=2^4-7$	62	0	0	0	0	82
−6	$1^3+1^3=2^3-6$	747	0	0	0	0	1 387
−5	$2^5+3^3=2^6-5$	14	0	0	0	0	14
−4	$3^3+1^3=2^5-4$	54	0	0	0	0	74
−3	$2^7+5^3=2^8-3$	14	0	0	0	0	16
−2	$5^3+1^3=2^7-2$	80	2	4	0	0	124
−1	$2^4+2^6=3^4-1$	38	4	0	0	0	60
0		**0**	**0**	**0**	**0**	**0**	**0**
1	$2^7+6^3=7^3+1$	86 106	14 952	10 904	8 822	7 936	902 336
2	$6^3+3^7=7^4+2$	4	0	0	0	0	4
3	$2^3+3^3=2^5+3$	16	0	0	0	0	16
4	$1^3+2^7=5^3+4$	58	0	0	0	0	78
5	$2^3+5^3=2^7+5$	9	0	0	0	0	9
6	$1^3+2^5=3^3+6$	64	0	0	0	0	84
7	$5^3+5^3=3^7+7$	11	0	0	0	0	13
8	$2^3+1^3=1^3+8$	1 554	264	188	148	124	12 114
9	$1^3+2^4=2^3+9$	88	0	0	0	0	116
10	$2^6+3^3=3^4+10$	10	0	0	0	0	10

Iz ove i gornjih tablica te ostalih rezultata dobivenih računalom možemo zaključiti sljedeće:

1. Naš program u programskom jeziku C++ pronalazi Pitagorine trojke već u prvih 100 prirodnih brojeva, dok Fermatove trojke ne nalazi niti među prvih milijun ili milijardu prirodnih brojeva. Ni Bealova jednadžba nema rješenje ako gledamo potencije do 10^{12} (prvih bilijun prirodnih brojeva), pa ako zaključujemo na osnovi Pitagorine i Fermatove jednadžbe, te induktivno iz tablice 3 (ako ih nema do tog broja, neće ih biti ni dalje) možemo reći da Bealova jednadžba nema rješenja i da se okrenuti treba prema dokazu nepostojanja brojeva A, B, C, x, y, z s traženim svojstvima [4].
2. Promatrajići interval od 1 do 5 milijardi u tablici 3 (s obzirom na najveću vrijednost, C^z) većina se vrijednosti za p pojavi već u intervalu od 1 do milijardu. Nakon toga, od milijardu (10^9) do pet milijardi, pojavljuju se samo promašaji veličine 1 i 8 te poneki

promašaj veličine -2 ili -1.

3. Promašaji veličina od -10 do $+10$ pojavljuju se sve rjeđe (promatrajući intervale veličine milijardu). To je logično jer u prvi interval, od 1 do milijardu, upada jako puno različitih potencija oblika x^y, gdje su x i y neki prirodni brojevi. Promatramo li npr. bazu 2, od 1 do milijardu pojavljuje se ukupno 19 potencija oblika 2^x, a od milijardu do dvije milijarde samo jedna: $2^{20} = 1\ 048\ 576$. Od dvije milijarde do tri milijarde također postoji samo jedna potencija broja 2, a od tri milijarde do četiri milijarde nijedna. Od četiri milijarde do pet milijardi opet susrećemo samo jednu potenciju broja dva, 2^{22}. Potencije bilo kojeg broja su „rijetke” u „intervalima velikih brojeva”. Koliko god velike intervale $[1, D]$, $[D, 2D]$, $[2D, 3D]$, … promatrali, počevši od 1 pa nadalje, potencije nekog broja sve će se rjeđe pojavljivati u tim intervalima. Promatramo li potencije broja 2 kojih u prvih milijardu brojeva ima 19, onda, ako želimo promatrati sljedeći interval u kojemu ih ima 19, moramo odabrati interval od milijarde do 274 milijarde, a sljedeći bi period bio od te 274 milijarde do $144 \cdot 10^{15}$ ili do 144 „milijuna milijardi”. Promatramo li potencije broja 10, morali bismo uzeti još veće nadolazeće intervale, a za veće brojeve od 10 još veće itd.

Pogledajmo koliko brojeva oblika x^y, gdje je y veći ili jednak 3, ima u prvih šest intervala veličine 1 000.

Tablica 4. Zastupljenost potencija oblika x^y, $y \geq 3$, u navedenim intervalima

1–1 000	1 000–2 000	2 000–3 000	3 000–4 000	4 000–5 000	5 000–6 000
2^3=8	2^{10}=1 024	2^{11}=2 048	5^5=3 125	2^{12}=4 096	18^3=5 832
2^4=16	6^4=1 296	3^7=2 187	15^3=3 375	17^3=4 913	
3^3=27	11^3=1 331	13^3=2 197			
2^5=32	12^3=1 728	7^4=2 401			
2^6=64		14^3=2 744			
3^4=81					
5^3=125					
2^7=128					
6^3=216					
3^5=243					
2^8=256					
7^3=343					
2^9=512					
5^4=625					
3^6=729					

1 050 000–1 180 000
(u ovom intervalu veličine 130 000 nema nijedne potencije oblika x^y, $y \geq 3$.)

Brojevi koji su potencije vrlo se rijetko pojavljuju u intervalima u području „velikih brojeva” jer što je baza veća, broj pojava njenih potencija gotovo iščezava u odabranim intervalima fiksne veličine, budući da ti brojevi eksponencijalno rastu, a granice intervala rastu linearno, tj. ako je prvi promatrani interval od 1 do x, drugi je od x do $2x$ itd. Zbog eksponencijalnog rasta morali bismo u potragu za pogodnim potencijama ići po eksponencijalno rastućim granicama intervala, što je i na računalu vrlo teško ako želimo istraživati „veće” brojeve.

4. U traženju brojeva koji zadovoljavaju Bealovu hipotezu koristimo se u prvih 1 000 prirodnih brojeva, po tablici 4, s točno 15 brojeva. Želimo li sada uzeti sljedećih 15 brojeva za kombinaciju s prvih 15, za sljedećih 15 moramo gledati interval od 1 000 do 7 000, zatim za sljedećih 15 brojeve od 7 000 do 21 000, i tako dalje. U traženju brojeva koji bi ispunili Bealovu jednadžbu, za još nekoliko „kandidata” koje želimo kombinirati s već pronađenim potencijama moramo često otići „daleko u velike brojeve”.
5. Od 1 000 000 do 2 000 000 postoje samo 33 broja koja možemo iskoristiti kao pribrojnike za Bealovu jednadžbu, no njihova je međusobna udaljenost u prosjeku oko 30 000, a najmanja udaljenost je od $110^3 = 1\ 331\ 000$ do $34^4 = 1\ 336\ 336$ jednaka 5 336 što znači da ako želimo koristiti ove brojeve u rješavanju Bealove jednadžbe neće nam biti od pomoći „blagi” razmaci među potencijama koji se nalaze (tablica 4) među prvih 5 000 prirodnih brojeva.
6. Unatoč tome što nas ovi argumenti vode prema zaključku da Bealovih trojki nema, rješenje jednadžbe $a^3 + b^3 = c^3 - 3$ navedeno gore pronađeno je tek među brojevima većima od 10^{21}, što nam sugerira oprez pri bilo kakvim radikalnim zaključivanjima o postojanju ili nepostojanju rješenja Bealove jednadžbe samo na osnovi situacije kod „manjih” brojeva.

6. Zaključak

Među velikim brojevima mogu se pojaviti brojevi svakakvih oblika pa nema (vidljivog) smisla odbaciti tvrdnju da negdje među njima postoje tri potencije, A^x, B^y i C^z za koje vrijedi $A^x + B^y = C^z$.

Pitanje postoji li rješenje Bealove jednadžbe može se predočiti pitanjem postoji li riba koja je potpuno crvena, s bijelim točkama? U prvih deset metara dubine situacija je istražena i tu takve ribe nema (to su npr. brojevi do 100, 1 000 ili 1 000 000). Dalje, kako idemo prema velikim dubinama, broj riba je sve manji i šanse da se uopće pronađe živo biće, a pogotovo koje je riba, su sve manje, no dubine su kod nas beskonačne pa tu možda među raznim, čudnim, (i nekim gorostasnim) bićima postoji i tražena ribica.

Traženje brojeva A, B, C, x, y i z koji bi zadovoljavali jednadžbu $A^x + B^y = C^z$, gdje su baze A, B i C relativno prosti brojevi, a eksponenti x, y i z veći ili jednaki 3 na računalu, može se predstaviti traženjem planeta na kojem kao i na zemlji ima života. Ako takav planet i postoji, za sada znamo da je vrlo daleko. Pravila su sljedeća: do tog planeta možemo doći brzo, gotovo odmah, ali svakako treba sletjeti nakon toga na njega i provjeriti ima li života na njemu, što nam troši vrijeme (u programskom jeziku Python možemo, na primjer, odmah odabrati neke velike brojeve A, B, C, x, y i z, ali sigurno će onda trebati vremena za računanje potrebnih potencija te njihovo zbrajanje i usporedbu). Kako se udaljavamo od Zemlje, broj planeta je sve manji (ako pretraživanje svemira organiziramo tako da se u koncentričnim sferama širimo od našeg planeta). Jedna od taktika je odabrati veliki C^z, i onda pokušati A^x od 1 povećavati u petlji do C^z i za taj A^x naći direktno koliki mora biti B^y, tj. odabrati točku u svemiru i pretražiti cijelu ravninu koja je sadrži. Druga taktika je odabrati točku u svemiru i oko nje onda istraživati, tj. naći A^x, B^y i C^z koji otprilike zadovoljavaju jednadžbu i onda pokušati promijeniti malo svaki od ta tri člana kako bi se približili jedan drugom (možda slučajno pronađemo traženi planet u blizini mjesta slučajno odabranih koordinata).

Od programerskih alata, za naše izračune najbolji se pokazao C++ (koji je sam po sebi naravno brži od Pythona, ali za razliku od Pythona najveći broj do kojega možemo računati je „samo” 2^{64}). C++ uz pogodnu paralelizaciju može unutar nekoliko sati provjeriti Bealovu jednadžbu za sve prirodne brojeve do maksimalno 2^{64} koliko iznosi granica tipa podataka long long int. Dakle, ako želimo rješenje gore navedenih problema tražiti na računalu moramo donijeti odluku hoćemo li algoritam pisati u programskom jeziku Python koji već ima ugrađene biblioteke za velike brojeve ili u programskom jeziku C (C++) koji je brži i u kojemu se programi jednostavnije daju paralelizirati, ali u koji moramo ugraditi dodatnu biblioteku za računanje s velikim brojevima (ako želimo račun s brojevima većima od 2^{64}) .

Na kraju predlažemo i brzi algoritam za traženje rješenja Bealove jednadžbe. Iako se u Bealovoj pretpostavci radi o 6 prirodnih brojeva, pokazuje se da je pametnim redoslijedom postavljanja varijabli neke petlje u pretragama moguće izbjeći. Na početku je najpraktičnije odabrati C^z. Tada, budući da zbroj $A^x + B^y$ mora biti jednak C^z, bar jedan od pribrojnika A^x i B^y mora biti broj veći od polovine broja C^z. Bez smanjenja općenitosti možemo odabrati A^x kao većeg od dva pribrojnika i onog koji je između polovine broja C^z i broja C^z. Sada, za odabrani A (petlja za A ide od 2 do $\sqrt[3]{C^z}$) postoji najviše jedan x sa svojstvom $0.5\,C^z < A^x < C^z$. Na kraju, kada smo odabrali C^z i A^x, znamo da je

$0 < B^y < C^z - A^x$, pa je dovoljno B uzimati u petlji od 1 do $\sqrt[3]{C^z - A^x}$, a y računati direktno kao $\log_B(C^z - A^x)$.

Čak i na najbržim računalima, uz sva algoritamska i strojna ubrzanja i razne načine paralelizacije algoritama, poradi tehničkih ograničenja bilo kojeg današnjeg i budućeg računala te zbog pretraživanja u „dalekom svemiru" velikih brojeva (udaljenosti od poznatih do sljedećih nepoznatih kandidata za ispunjavanje Bealove jednadžbe nam rastu eksponencijalno), traženje šestorke rješenja Bealove jednadžbe i dalje ostaje težak zadatak.

Literatura

[1] `http://www.ams.org/profession/prizes-awards/ams-supported/beal-conjecture` (zadnji pristup travanj 2020.)

[2] A. Dujella, *Uvod u teoriju brojeva (skripta)*, `http://e.math.hr/zeta/utblink.pdf` (zadnji pristup travanj 2020.)

[3] R. Daniel Mauldin, *A Generalization of Fermat's Last Theorem: The Beal Conjecture and Prize Problem*, Notices of the AMS. (1997) **44 (11)** 1436–1439.

[4] `https://en.wikipedia.org/wiki/Beal_conjecture` (zadnji pristup travanj 2020.)

Josip Ćatić
Elektrotehnička i prometna škola Osijek

Davor Damjanović
student FERIT-a Osijek

Filip Đuričković
student FERIT-a Osijek

Leon Gal
student FERIT-a Osijek

Zvonimir Haramustek
III. gimnazija Osijek

Leon Kodžoman

student FERIT-a Osijek

Josip Koprivnjak
Elektrotehnička i prometna škola Osijek

Borna Krušlin
III. gimnazija Osijek

David Majdandžić
student FERIT-a Osijek

Neven Marić
Elektrotehnička i prometna škola Osijek

Luka Miler
Elektrotehnička i prometna škola Osijek

Tomislav Rudec
docent na FERIT-u, Osijek
E-mail adresa: `trudec@ferit.hr`

Krešimir Turkalj
student FERIT-a Osijek

Ivan Vazler
docent na Odjelu za fiziku, Osijek