Prostost

Fran Vojković i Alen Živković

Naš će se projekt baviti prostosti i testiranjem iste. Korisničko sučelje je jednostavno: korisnik može upisati jedan broj te stisnuti tipku PROVJERI nakon čega aplikacija koristi nekoliko načina za provjeravanja prostosti upisanog broja te ispisuje rezultate za svaki. Planirani testovi zasad su: jednostavna metoda dijeljenja do korijena, Wilsonov teorem, Miller-Rabinov test i Solovay-Strassenov test. Nije isključeno da dodamo još neku metodu. Želimo da korisnik može u bilo kojem trenutku izračunavanja za posljednja dva testa prikazati takozvano *step by step* rješenje. Odnosno, za Miller-Rabinov test se obično provjerava barem 20 baza pa da korisnik može vidjeti koje su to baze nasumično odabrane te kako je protekla provjera. Isto i za Solovay-Strassenov test, samo za veći broj baza (između 50 i 100).

Naravno da će za vrijeme provođenja testova aplikacija implicitno za određene unesene brojeve moći odrediti jesu li Euler pseudoprosti u nekoj bazi. Zato želimo u bazi podataka čuvati 100 brojeva koji predstavljaju baze, a aplikacija mora u bilo kojem trenutku kada dobije da je neki broj Euler pseudoprost u toj bazi spremiti taj broj na pripadajuće mjesto u bazi. Korisnik onda preko sučelja može otvoriti tablicu Euler pseudoprostih brojeva koja će se ispunjavati postepeno ovisno o tome za kakve brojeve provjerava prostost. Veličina tablice, odnosno koliko Euler pseudoprostih brojeva želimo imati mogućnost spremiti za svaku bazu, ćemo naknadno odrediti.

Korisnik pri unosu broja može otvoriti napredne opcije koje mu dozvoljavaju da dozvoli samo neke određene testove, a ostale zanemari. Također može odabrati na koliku vjerojatnost je voljan pristati u slučaju odabira vjerojatnosnih testova (Miller-Rabinov i Solovay-Strassenov) što će onda mijenjati broj baza na koje će ovi testirati njegov uneseni broj. Trenutno razmatramo još neke napredne opcije poput prikaza vremena potrebnog za odrađivanje testa.

Testiranjem ćemo utvrditi je li potrebno za dovoljno velike brojeve zabraniti upotrebu klasičnih algoritama te dopustiti samo dva vjerojatnosna testa. Pokušat ćemo za neke brojeve dodati opciju provjeravanja radi li se o Carmichaelovom broju pa bi u tom slučaju mogli slagati i takvu tablicu u bazi podataka. Također, korisnik će moći zadati samo broj i bazu te dobiti odgovor je li taj broj pseudoprost i Euler pseudoprost u toj bazi (gdje se možemo poslužiti bazom podataka).

Po mogućnosti, možemo dodati i usporedbe brzine određenih testova s obzirom na veličinu broja za kojeg treba testirati prostost. Pokušat ćemo testove učiniti pogodnima i za jako velike brojeve, uglavnom efikasnom implementacijom modularnog potenciranja.