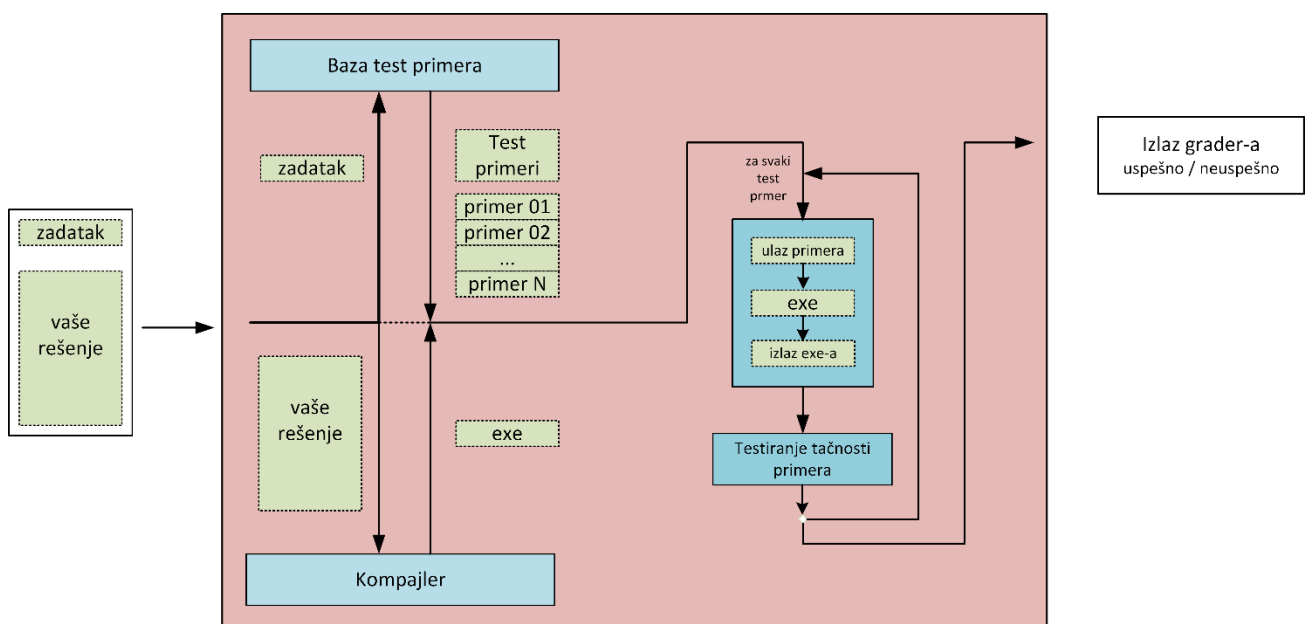


Grader

Grader predstavlja sistem za automatsko testiranje rešenja algoritamskih problema. Kako je testiranje automatizovano, svi protokoli moraju biti formalno i strogo definisani.

Sama procedura testiranja je sledeća:

- Korisnik pošalje (eng. submit) rešenje problema i označi u kom programskom jeziku je poslao kod.
- Grader prihvata zahteva za testiranje. Zahtev se može opisati četvorkom (korisnik, zadatak, kod, programski jezik).
- Grader zatim kompajlira primljeni kod, za dati programski jezik, čime se dobija izvršni program koji treba testirati.
- Svaki zadatak sadrži niz test primera. Za svaki od test primer grader će odraditi sledeće:
 - Pokrenuti korisnikov program.
 - Na standardni ulaz prosleđuje ulaz trenutnog test primera (input).
 - Ukoliko korisnikov program radi duže od datog vremenskog ograničenja, program se prekida i test primer se proglašava neuspešnim.
 - Vršiti se proveru memorijskog ograničenja: ukoliko je program iskoristi više od dozvoljene memorije, test primer se proglašava neuspešnim.
 - **Inače, grader uzima njegov rezultat (koji je dat preko standardnog izlaza).**
 - Zatim se proverava tačnost rešenja.
 - Ukoliko je rešenje tačno, test primer se proglašava upešnim; u suprotnom proglašava se neuspešnim.
- Da bi zadatak bio prihvaćen, korisnikov program mora uspešno rešiti sve test primere.



Slika 1. Grafički prikaz rada grader-a

Boldovana stavka u proceduri testiranja je najbitnija. Kako bi se moglo vršiti automatsko testiranje, vaš program mora poštovati formate ulaza i izlaza (koji su strogo definisani u svakom problemu). Primera radi, ukoliko na kraju rezultat, koji je jednak 5, štampate uz rečenicu: "konacno resenje je: 5", test primer neće biti uspešan bez obzira da li je 5 tačno rešenje ili ne. Razlog za ovo je što će grader upoređivati ta broj 5 (tačno rešenje) sa vašim rešenjem koje je "konacno resenje je: 5". Tačnije ovo rešenje neće proći ni samo parsiranje u ceo broj za ovaj primer.

Mnogi će reći da su ovo nebitne greške koje nemaju veze sa samim algoritmom, što je tačno, i kao takve ukoliko do njih dođe treba ih ispraviti i samo rešenje prihvatiti. Možda zvuči surovo ali ovakva pravila moraju poštovati. U suprotnom vrlo brzo bi došlo do pitanja gde onda povući granicu za menjanje koda, pošto de se dobijati svakakve žalbe šablona: ukoliko se u mom kodu na tom i tom mestu promeni karakter x u y moj algoritam bi bio korektan.

Nadamo se da smo vam na ovaj način malo približili rad grader-a, odnosno proceduru automatskog testiranja. Ukoliko imate dodatnih komentara / pitanja, obratite se BubbleBee timu.