

Predgovor

„Genijalnost je u svemu intuicija.

Sva značajna dela ispod tog fenomena tvorevine su talenta.“

Balzak

Danas je verovatno malo ljudi koji se bave programiranjem, a da nisu čuli za objektno orijentisano programiranje. Više je onih koji su za objektno orijentisano programiranje i čuli, ali o njemu malo znaju. Ova knjiga ima za cilj da svima njima pomogne, kako bi što lakše ušli u svet objektno orijentisanog programiranja na jeziku C++.

Jezik C++ je danas svakako najpopularniji objektno orijentisani jezik. To je veoma moćan, fleksibilan jezik, udoban za programiranje. Programeri su ga veoma rado i brzo prihvatili, baš kao i njegovog prethodnika – jezik C. Uzrok ovakvog uspeha verovatno leži, između ostalog, i u tome što je jezik C++ projektovao jedan programer, prilagođavajući ga svom ukusu i potrebama.

Nažalost, C++ nije nimalo jednostavan jezik. Ponekad se čoveku učini da je to jezik bez granica i da ga ne može lako sagledati u celini. Zbog toga se ne treba brinuti. Najzad, nije važno znati *sve* detalje jednog tako složenog jezika kao što je C++, već je najvažnije znati *pravilno upotrebljavati* one delove koji su u datom trenutku potrebni. Praksa je ta koja čoveka uči da na najbolji način upotrebi alat za postizanje željenog cilja. Ipak, oprezan čovek uči na tuđim iskustvima. Ova knjiga ima namenu i da prenese deo iskustva iz oblasti objektno orijentisanog programiranja na čitaoca, kako ne bi pravio greške koje se mogu sprečiti pažljivijim programiranjem.

Knjiga je nastala kao rezultat dužeg rada autora na projektima sistemskog i aplikativnog programiranja, kao i na obrazovanju u oblastima objektno orijentisanog programiranja i jezika C++. Autor se iskreno nada da lagani, postupni i potpuni pristup ovoj oblasti može pomoći čitaocu u temeljnom savlađivanju problematike.

Svi naponi su učinjeni da se u knjizi ne pojave greške. Ipak, sve ispravke i sugestije čitaoca autoru samo su dobrodošle.

Kome je knjiga namenjena?

Knjiga je namenjena svima onima koji žele da dobro nauče jezik C++ i osnovne principe objektno orijentisanog programiranja, kako bi ih primenjivali u praksi.

Knjiga će prvenstveno biti korisna profesionalnim programerima koji rade samostalno ili u timovima, i koji realizuju *velike* softverske projekte. Savladavanjem objektno metodologije kao strategije i jezika C++ kao sredstva, profesionalni programeri će povećati efikasnost svoga rada na složenim sistemima. Potrebno je naglasiti da se koncepti objektno orijentisanog programiranja primenjuju podjednako uspešno kako na području aplikativnog, tako i na području sistemskog programiranja.

Sve je očiglednije da su objektno orijentisano programiranje i jezik C++ opredeljenje za budućnost. Zato ova knjiga može biti prigodan udžbenik za sve one koji nemaju mnogo iskustva u programiranju, a rešeni su da odmah krenu na „pravu stranu“.

Najzad, ova knjiga može biti od koristi i iskusnijim programerima koji već znaju ponešto o objektno orijentisanom programiranju i jeziku C++, jer sadrži delove koji iznose stilska uputstva, iskustva, kao i opis jedne metodologije objektno orijentisanog projektovanja.

Knjiga obuhvata potpun pregled jezika C++ kao samostalnog jezika i ne razmatra njegove sličnosti i razlike sa jezikom C. Zato je može čitati i onaj ko ne zna ništa o jeziku C, ili o njemu zna malo. Znanje jezika C je na mnogim mestima korisno, ali, po iskustvu autora, može biti i otežavajući faktor pri prelasku na jezik C++. Prema tome, čini se da su u približno istoj poziciji i oni koji znaju jezik C, i oni koji ga ne znaju.

Ova knjiga nije opšti udžbenik iz programiranja. Za prve korake u programiranju preporučuje se neka druga literatura iz oblasti strukturiranog programiranja [2,7,9]. Smatraćemo ovde da su čitaocu jasni svi osnovni koncepti programiranja: tipovi podataka, promenljive, izrazi, naredbe i potprogrami (zajedno sa mehanizmima prenosa argumenata). Iako ova knjiga detaljno obrađuje sve navedene koncepte u jeziku C++, pa i apsolutni početnik može da se usudi da uči programiranje iz nje, ipak je korisno da čitalac poznaje osnovne koncepte iz nekog drugog, tradicionalnog programskog jezika. Preporučuje se znanje jezika Pascal, prema referenci [9].

„Šta je objektno orijentisano programiranje? Kako je i *zašto* nastalo? Kako se koriste principi objektno orijentisanog programiranja u praksi? Zašto toliko priče oko objektno orijentisanog programiranja? Zar su sve te stvari toliko teške, da se ne mogu lako savladati? Kako se, najzad, pišu *pravi* programi u objektno orijentisanom duhu, koji, pored toga što su *objektno orijentisani*, uz to još i *rade*? Zašto bi neko napuštao stari, dobri svet tradicionalnog programiranja, ako je tako dobro i lako pisao programe na stari način?“

Ako su čitaoca ikad mučila ovakva i slična pitanja, verujemo da će mu ova knjiga dati odgovore na njih, kao i na mnoga druga.

Ako je čitalac rešen da postane *pravi*, ozbiljni programer, ili se time već smatra, ova knjiga je njemu namenjena. Ako čitalac nema nameru da se ozbiljno bavi programiranjem, ili mu programiranje treba samo za skromne programerske poduhvate, nije potrebno da čita ovu knjigu. Bolje je da potraži neko lakše štivo.

Sadržina knjige

Knjiga nosi podnaslov „Od osnovnih pojmova do naprednijih tehnika“ i prilagođena je potpunim početnicima u oblasti objektno orijentisanog programiranja i jezika C++. Sadržaj iscrpan opis ovog jezika, uključujući i koncepte koji su „nasleđeni“ iz jezika C. Ova knjiga takođe sadrži i elemente objektno orijentisanog projektovanja i iskustvene, stilske napomene, koje se već mogu smatrati „naprednijim tehnikama“.

Knjiga se zaustavlja tamo gde počinju mnoge druge knjige iz ove oblasti. Ona se ne vezuje ni za jednu konkretnu biblioteku klasa, okruženje ili prevodioca. Završava se tamo gde se, po proceni autora, završava jezik C++ i uvod u objektno orijentisano programiranje, a počinje dug put praktične eksploatacije, upoznavanja sa već napisanim softverom itd.

Hijerarhijska podela celina u knjizi nosi sledeće nazive: deo, glava, poglavlje i odeljak. Oni će se u knjizi koristiti za upućivanje na pojedine celine.

Knjiga se sastoji iz četiri dela. U prvom delu postupno su prikazani osnovni razlozi za uvođenje objektno orijentisanog programiranja i osnovni koncepti jezika C++. Dat je samo površan i globalan prikaz kako bi čitalac stekao osnovnu sliku o celoj temi. Verujemo da ovakav pristup pomaže lakšem razumevanju ostalih delova knjige, koji se bave detaljima.

Drugi deo sadrži opis delova jezika C++ koji nisu u direktnoj vezi sa objektno orijentisanim pristupom. Ovi delovi su uglavnom nasleđeni iz jezika C, ali ima i onih koji su prvi put uvedeni u jezik C++.

U trećem delu je dat detaljan opis svih objektno orijentisanih koncepata jezika C++. U ovom delu prikazani su i mehanizmi generisanja klasa i obrade izuzetaka.

Četvrti deo sadrži uvod u metodologiju objektno orijentisanog projektovanja softvera. Tu su data osnovna uputstva za dalji praktičan rad.

U knjizi je namerno primenjen pristup jeziku C++ „sa strane“ objektno orijentisanih koncepata. Autor smatra da je C++ prvenstveno namenjen objektno orijentisanom programiranju i da su, stoga, ključni koncepti klase i nasleđivanja, koji predstavljaju model, a ne ugrađeni tipovi podataka, izrazi ili naredbe, koji predstavljaju implementaciju. Zato su najpre načelno objašnjeni objektno orijentisani koncepti da bi se čitalac navikao da razmišlja „u obrisima“ objekata i da implementaciju shvati kao lakši deo posla u programiranju od samog projektovanja. Ovakav pristup svakako je lak za one koji već znaju jezik C, ali se autor nada da je podjednako dobar i za one koji nemaju mnogo iskustva u programiranju.

Opis jezika C++ bazira se strogo na referenci [8]. Pojedini delovi knjige, uglavnom specifikacije pravila jezika, preuzeti su iz ove reference. Knjiga se ne oslanja ni na jednu posebnu implementaciju prevodioca, niti na neki poseban računar ili okruženje. Treba napomenuti da različite implementacije prevodilaca za jezik C++ sadrže manja ili veća odstupanja od pravila navedenih u ovoj knjizi. U svakom slučaju, ova odstupanja će polako nestajati sa novim verzijama prevodilaca, a verovatno i sasvim nestati kada jezik C++ bude standardizovan.

Najzad, treba reći i šta knjiga *ne* sadrži. Ona ne sadrži preciznu specifikaciju jezika, koja je ionako potrebna samo realizatorima prevodilaca. Za preciznu specifikaciju jezika čitalac se upućuje na referencu [8]. Knjiga takođe ne sadrži opis nekih gotovih biblioteka klasa, iako se profesionalno objektno orijentisano programiranje ne može zamisliti bez upotrebe tuđeg, gotovog koda. Ako čitaoca ovo zanima, treba da pogleda npr. referencu [5]. Na kraju, knjiga ne sadrži detaljan opis neke metodologije objektno orijentisanog programiranja, kojih ima više. Za to su specijalizovane neke druge knjige iz objektno orijentisanog projektovanja (npr. [3]).

Kako čitati knjigu?

Ako čitalac ne zna jezik C i početnik je u objektno orijentisanom programiranju, preporučujemo da knjigu čita redom, od početka do kraja.

Ako čitalac zna jezik C, a početnik je u objektno orijentisanom programiranju, može samo površno preći opise delova jezika C++ koji su nasleđeni iz jezika C. Ipak, nije preporučljivo da preskače cele glave, osim glave 3, jer su, zbog celovitosti, originalni koncepti jezika C++ rasuti po celoj knjizi.

Ako čitalac nije početnik u objektno orijentisanom programiranju, ali ne zna jezik C++, može preskočiti glavu 17.

Knjiga se može koristiti i kao udžbenik ili priručnik za kurs iz objektno orijentisanog programiranja na jeziku C++. U tom slučaju predavač može napraviti izbor poglavlja koja će predavati. Za kurs koji obuhvata celu knjigu potrebno je oko 60 časova.

U svakom slučaju, dobro je poštovati redosled izlaganja u knjizi. Posebno je važno razumeti sadržaj glava 1 i 2, jer se tu izlaže sama suština objektnog pristupa i jezika C++.

Knjiga sadrži veliki broj programskih primera. Ovi primeri nisu vezani ni za jednu implementaciju prevodioca posebno, pa zato nisu davana posebna uputstva vezana za unošenje programa, prevođenje, povezivanje i izvršavanje. Smatramo da se čitalac može lako i sam snaći sa okruženjem koje poseduje. Preporučujemo da kompletne primere u knjizi čitalac izvrši na računaru.

Na početku gotovo svakog odeljka nalazi se jasno istaknut okvir sa rezimeom tog odeljka. Ovaj rezime dat je u obliku kratkih rečenica, taksativno i eksplicitno. Ovi okviri imaju dvojaku ulogu. Prvo, tokom prvog čitanja knjige, čitaocu se preporučuje da tekst iz okvira pročita ne pokušavajući da razume sve detalje. Onim čitaocima koji detaljno poznaju druge programske jezike, naročito jezik C ili neki drugi objektno orijentisani jezik, ovi navodi će pomoći utoliko što odmah iznose najvažnije činjenice i rešenja određenih koncepata u jeziku C++. Neki okviri sadrže i jasne podele i specifikacije pojmova koji su u tekstu postupno i detaljno objašnjeni. Dakle, tokom prvog čitanja, ovi okviri svim čitaocima služe najpre kao orijentir. Drugo, posle prvog čitanja, ovi okviri će čitaocu poslužiti kao kratak podsetnik. Može se reći da se pregledom ovih okvira, posle prvog detaljnog čitanja, knjiga može ponovo pročitati za nekoliko sati uz poimanje ključnih delova. Ovi okviri često imaju izvode iz gramatike jezika C++. Oni su preneseni u originalu i često nisu razumljivi, mada su sasvim precizni. Razumljivije objašnjenje sintakse nalazi se u tekstu, a potpuni pregled gramatike dat je u Prilogu 1. Jedino okviri prvog dela knjige sadrže neformalne specifikacije sintakse koje nisu kompletne.

Na pojedinim mestima dati su pregledi određenih koncepata ili složenih pravila jezika. Ovi pregledi služe za bolje razumevanje detaljnih objašnjenja iz teksta, a ponovljeni su u Prilogu 2, tako da ih čitalac može lako naći i pri upućivanju sa drugih mesta u knjizi. Pojedini pregledi sadrže algoritme odgovarajućih postupaka prevodioca, iskazane formalno, u pseudojeziku koji liči na Pascal. Nikakve posebne specifikacije ovog pseudojezika nisu date, jer će zapisi algoritama nesumnjivo svakom čitaocu biti jasni.

Prilog 3 je spisak literature korišćene za pisanje ove knjige. Brojevi u uglastim zagradama [] u tekstu označavaju literaturu na koju se čitalac upućuje. Prilog 4 sadrži rečnik originalnih engleskih termina tematike ove knjige. Ovaj rečnik ima za cilj da čitaocu pomogne u pronalaženju srpskog prevoda nekog opšte usvojenog engleskog termina koji se koristi u ovoj knjizi. Posle pronalaženja srpskog prevoda, čitalac može pronaći definiciju ili objašnjenje datog (srpskog) termina pomoću indeksa u Prilogu 5.