
Uvod

Nov pristup programiranju na jeziku C++

Verovatno želite da što brže naučite da pišete korisne programe na C++-u. Zato vas prvo upoznajemo s najkorisnijim elementima jezika C++. Uobičajeno je da pre C++-a naučite njegovu osnovu, jezik C. Mi smo radikalniji i od samog početka kursa bavimo se složenim strukturama podataka čije ćemo poreklo objašnjavati kasnije. Za vrlo kratko vreme moći ćete da programirate na C++-u po predloženom uzoru.

Naš pristup je specifičan i po tome što nas od istraživanja jezika i njegovih biblioteka više zanima rešavanje problema. Umesto da programe koristimo kao objašnjenje pojedinih odlika jezika, jeziku pristupamo samo kao sredstvu za pisanje programa.

Knjiga je posvećena programiranju na C++-u više nego karakteristikama ovog jezika i biće naročito korisna čitaocima koji poznaju C++ i žele da ga uspešnije upotrebljavaju. Početnici često nauče samo tehničku stranu C++-a, ali ne i kako da ovaj jezik primenjuju za rešavanje svakodnevnih problema.

Za početnike i iskusne programere

Neko vreme smo na Univerzitetu Stanford svakog leta držali jednonedeljni intenzivni kurs C++-a. Pretpostavljajući da polaznici poznaju C, na početku smo ih učili definisanju klasa, a zatim sistematično i ostatku jezika. Polaznici su isprva bili zbunjeni i obeshrabreni, ali se situacija popravljala pošto bi naučili dovoljno da mogu napisati korisne programe. Posle toga, sve je išlo glatko.

Kurs smo unapredili predavanjima o standardnoj biblioteci. Polaznici su sada od samog početka učili kako da se služe bibliotekom i pišu korisne programe, a detalje o tome smo im otkrivali tek kada bi naučili toliko da mogu da ih koriste produktivno.

Napredak je bio neverovatan jer su polaznici već drugog dana po početku kursa mogli da pišu programe za koje bi im po starom kursu trebala čitava prva nedelja. Više niko nije bio zbunjen ni obeshrabren.

Sažetak

Jezik C++ je imao dovoljno vremena da „sazri“, a u skladu s tim se i naše razumevanje ovog jezika razvijalo. Sada možemo da zanemarimo mnoge osnovne ideje koje su bile glavni oslonci ranijim programima i programerima na C++-u. Zbog toga je prethodno opisani pristup učenju C++-a moguć.

Samo zrele tehnologije dozvoljavaju zanemarivanje detalja. Prvi modeli automobila su se, na primer, tako često kvarili da je svaki vozač morao da poznaje automehaničarski zanat. Bilo je nepromišljeno otputovati i rizikovati da se ne vratite ako vam se dogodi nezgoda sa automobilom. Vozači danas ne moraju detaljno da poznaju mehaniku automobila da bi ih vozili. Tehničkim detaljima možete se baviti iz zadovoljstva ili iz drugih razloga, što nas ovde ne interesuje.

Apstrakcija, odnosno selektivno zanemarivanje, podrazumeva usredsređivanje na ideje koje su važne za dati zadatak i zanemarivanje ostalog. Uvereni smo da je ova ideja najvažnija u modernom programiranju. Ključ za pisanje uspešnih programa je znati koje elemente problema uzeti u obzir, a koje zanemariti. Svaki programski jezik raspoložive alatkama za korisno apstrahovanje, koje uspešni programeri znaju da upotrebe.

Ova knjiga je puna apstrakcija, koje smatramo veoma korisnim. One mogu imati razne oblike, zbog čega i ne koristimo direktno reč „apstrakcija“. U knjizi, umesto toga, pominjemo i koristimo funkcije, strukture podataka, klase i nasleđivanje.

Mislimo da je pametno odabrane i dobro isplanirane apstrakcije upotrebljavati i ako ne znate detaljno kako funkcionišu. Kao što ne treba da budete automehaničar da biste vozili automobil, ne morate ni da znate u potpunosti kako radi C++ da biste programirali.

Čime se ova knjiga bavi

Ako ste ozbiljno odlučili da se bavite programiranjem na C++-u, moraćete sve iz ove knjige da naučite napamet, čak i ako ona ne govori o svemu što bi trebalo da znate.

Ovo što smo rekli nije besmisleno kao što zvuči. Nijedna knjiga ovog formata ne može da sadrži sve što ćete morati da znate o C++-u jer različitim programerima i raznim aplikacijama trebaju različita znanja. Knjige koje se bave svim aspektima C++, na primer Stroustrupova *The C++ Programming Language* (Addison-Wesley, 2000) govore više nego što treba da znate. Ali ono što vama ne koristi, zatrebaće nekom drugom programeru.

Međutim, pojedini delovi C++-a su globalno važni tako da morate da ih razumete ako hoćete da budete produktivni u programiranju. Takvim elementima C++-a smo posvetili pažnju. Moći ćete da napišete mnogo korisnih programa služeći se samo informacijama iz ove knjige. Jedan od naših recenzenata, vodeći programer važnog komercijalnog sistema napisanog na C++-u, rekao je da ova knjiga obuhvata sve elemente koje i on koristi prilikom programiranja.

Pomoću alatki iz ove knjige pišaćete prave programe na C++-u, ne u stilu C-a, ili nekog drugog jezika. Kada ovladate gradivom ove knjige, znaćete dovoljno da shvatite šta vas interesuje i kako da do toga dođete. Ljudi koji se amaterski bave sastavljanjem teleskopa imaju izreku da je lakše prvo napraviti 3-inčno ogledalo, pa tek onda 6-inčno, nego odmah praviti veće ogledalo.

Mi se bavimo samo standardnim C++-om, a sve nadogradnje zanemarujemo. Prednost ovakvog pristupa je u tome što će programi koje ćete naučiti da pišete raditi skoro svuda. Nećemo se, međutim, baviti programima namenjenim Windowsu jer su oni vezani za specifičnu podlogu, a često i za određenog prevodioca. Ako vas interesuje da pišete programe za specifično okruženje, moraćete to da naučite na nekom drugom mestu. To ne znači da vam ova knjiga neće koristiti. Naš pristup je univerzalan i sve što ovde naučite moći ćete da primenite u bilo kom okruženju. Nećemo vas sprečavati da pročitate, na primer, knjigu o programiranju grafičkih okruženja, ali vas molimo da pre toga proučite ovaj udžbenik.

Napomena iskusnim programerima koji koriste i C i C++

Kada učite nov programski jezik, možete doći u iskušenje da pišete programe u stilu jezika koje već poznajete. U ovoj knjizi se borimo protiv toga tako što vas podstičemo da od samog početka koristite složene apstrakcije iz standardne C++ biblioteke. Iskusni programeri koji koriste i C i C++ primetiće da su neke strane ovakvog pristupa u isto vreme i dobre i loše.

Iznenadićete se koliko malo će vam prethodno znanje pomoći da razumete C++ sa naše tačke gledišta. Neće vam se svideti to što ćete na početku morati da učite mnogo više nego što očekujete, ali dobra vest je da ćete napredovati neočekivano brzo. Ako ste C++ savladali oslanjajući se na prethodno stečeno znanje jezika C, verovatno ćete programe na C++-u pisati u stilu C-a. Nemamo ništa protiv takvog pristupa, ali naš koncept se razlikuje u toj meri da će vam se činiti da upoznajete stranu C++-a za koju niste znali.

Naravno, sintaksa dva jezika je slična, ali to su samo sitnice. Mi se važnim idejama bavimo redosledom koji ne očekujete. Na primer, pokazivače i nizove pominjemo tek u poglavlju 10, a popularne funkcije `printf` i `malloc` potpuno smo izostavili. S druge strane, klasom *string* iz standardne biblioteke bavimo se već u poglavlju 1. Naš pristup je, dakle, sasvim drugačiji.

Struktura knjige

Knjiga se uslovno može podeliti na dva dela. U prvih sedam poglavlja obrađujemo programe koji koriste apstrakcije iz standardne biblioteke. Drugi deo knjige, od osmog poglavlja do kraja, uči vas kako da sami pravite apstraktne oblike.

Nije baš uobičajeno da kurs programiranja započnemo lekcijama o C++ biblioteci, ali nama se čini da je to dobra ideja. Veći deo jezika C++, naročito elementi koji se teže savlađuju, namenjeni su autorima biblioteka. Korisnicima biblioteka takva znanja ne trebaju. Time što zaobilazimo ove elemente do drugog dela knjige, omogućavamo vam da korisne programe na C++-u počnete da pišete mnogo brže nego ukoliko biste se pridržavali uobičajenog pristupa.

Kada naučite da koristite biblioteku, upoznaćemo vas sa osnovnim sredstvima pomoću kojih se ona gradi i naučiti kako da sami pišete biblioteke. Pored toga, razvićete osećaj za uspešno pravljenje korisnih biblioteka bez suvišnog koda.

Iako je ova knjiga manja od većine knjiga koje se bave C++-om, trudili smo se da sve važne ideje upotrebimo u njoj najmanje dvaput, a one ključne i više puta. Zbog toga se mnogi delovi knjige pozivaju na druga mesta u knjizi. Primer za ovakva pozivanja je referenca u odeljku 7.4.3. Da bismo vam u skrenuli pažnju na novu ideju, za nju upotrebljavamo **polucrni kurzivni** stil.

Svako poglavlje (izuzev poslednjeg) ima odeljak *Detalji*. Ovo smo uveli da biste lakše zapamtili ideje iz datog poglavlja i upoznali se s dodatnim materijalom vezanim za njih koji treba da poznajete. Predlažemo da ove detalje pri prvom čitanju samo površno proučite, a po potrebi im se kasnije vratite.

U dva dodatka ukratko izlažemo i osvetljavamo važne delove jezika i biblioteke i bavimo se njihovim detaljima koji će vam koristiti prilikom pisanja programa.

Izvučite maksimum iz ove knjige

Kao i svaka knjiga namenjena programiranju, i ova sadrži primere programa. Najbolji način da shvatite kako ovi programi rade jeste da ih pokrenete na računaru. Neće vam biti problem da nađete odgovarajući računar, ima ih mnogo, a kako se novi pojavljuju skoro svakodnevno, sve što ovde o tome napišemo biće zastarelo ili netačno kada budete čitali. Ako ne znate kako se prevode i izvršavaju programi na C++-u, korišćiće vam Web lokacija <http://www.acceleratedcpp.com>. Ovu lokaciju ćemo povremeno osvežavati novim informacijama i savetima o strukturi i izvršavanju C++ programa. Ovde ćete naći i neke programe iz knjige i zanimljiva obaveštenja.

Zahvalnica

Želimo da zahvalimo ljudima bez kojih ova knjiga ne bi postojala. Mnogo dugujemo našim recenzentima: Robertu Bergeru, Dagu Brücku, Adamu Buchsbaumu, Stephenu Clamageu, Jonu Kalbu, Jeffreyu Oldhamu, Davidu Slaytonu, Bjarneu Stroustrupu, Albertu Tenbuschu, Bruceu Tetelmanu i Clovisu Tondou. Zahvaljujemo Tyrellu Albau-ghu, Buunyju Amesu, Mikeu Hendricksonu, Deborah Lafferty, Cathy Ohala i Simone Payment, kao i drugim ljudima iz izdavačke kuće Addison-Wesley koji su dali svoj doprinos. Alexandar Tsiris je proverio grčku etimologiju u odeljku 13.2.2. Na kraju moramo pomenuti brojne polaznike naših kurseva i predavanja koji su nas podstakli da početnike uvodimo u C++ preko složenih programa.

Andrew Koenig Gillette, New Jersey
Barbara E. Moo jun 2000