

Programski jezici i paradigme

Sanja Mitrović
mitrovic.sanja97@gmail.com

22. oktobar 2018.

1 Uvod

Jezik je osnovno sredstvo sporazumevanja ljudi. Koristeći jezik saznajemo nove informacije i delimo svoje znanje, osećanja i iskustva sa drugima. Jezikom možemo izraziti svaku misao i opisati svaki događaj. Svet se kontroliše kroz jezik. Svi koriste jezik za davanje uputstva drugima, kao i za prikupljanje informacija.

Jezik je neophodan i za računar. Softver se pravi pomoću specijalnih jezika koji izdaju naredbe računaru. Pored toga, jezik definiše podatke sa kojima će te naredbe raditi. Računarski jezik je sličan ljudskom jeziku. Imenice, glagoli, predlozi i objekti iz jezika imaju svoje odgovarajuće elemente u programskom kodu – redovima teksta koji se prevode u programe. Ali softverske rečenice imaju posebnu sintaksu, a reči od kojih se sastoji jezik imaju svoje precizno značenje.

2 Programski jezici

Ljudi za međusobnu komunikaciju koriste jezik, i to osnovno je sredstvo za predstavljanje i prenošenje informacija. Čovek i računar komuniciraju pomoću programskog jezika. Čovek mora na neki način da saopšti računaru niz instrukcija koje treba da izvrši. Programski jezik je sredstvo kojim čovek saopštava računaru program. Prirodni jezici dopuštaju nejednoznačnost i nepreciznost. Računar može „razumeti” samo formalni zapis, ne tolerišući ni najmanje nepreciznosti. Programski jezici računaru omogućavaju zapis niza instrukcija koje se ne mogu višeznačno protumačiti. Jednoznačnost svake konstrukcije programskog jezika je njegova najbitnija karakteristika. Programski jezik je skup pravila kojim se računaru predstavljaju instrukcije i opisuju podaci.

Programske jezike [4] možemo podeliti po stepenu zavisnosti programskog jezika i računara na:

- mašinski zavisne (jezici niskog nivoa, eng. "low-level")
- mašinski nezavisne(jezici višeg nivoa, eng. "high-level")

Mašinski jezik je izgrađen nad binarnom azbukom. Sve operacije i svi podaci predstavljeni su nizovima bitova. U zavisnosti od arhitekture računara, ti nizovi bitova imaju različita tumačenja i različite su veličine. Kako se mašinski program sastoji od niza nula i jedinica i zahteva dobro poznavanje načina rada

i arhitekture određenog računara, vrlo je teško programirati na njemu. Programi na prvim računarima bili su zapisani mašinskim jezikom što je uslovalo da uzak krug ljudi piše i održava programe. Da bi se premostile poteškoće pisanja programa na mašinskom jeziku, dolazi do razvoja simboličkih jezika. Umesto instrukcija pisanih nizom bitova, uvedene su mnemotehničke skraćenice za operacije i simboličke oznake podataka, npr. naredbom ADD a, b vrši se sabiranje podataka a i b. Na taj način proces programiranja je u znatnoj meri olakšan, ali i dalje zavisi od konkretnog procesora, tj. i dalje je potrebno poznavati tehničke karakteristike konkretnog računara za koji pišemo program. Da bi se program napisan na simboličkom jeziku izvršavao na računaru, mora se prethodno prevesti na mašinski jezik. Kako svakoj naredbi simboličkog jezika odgovara jedna naredba mašinskog jezika, posao je automatizovan tako što je napisan program koji kao ulaz dobija program napisan u simboličkom jeziku, a kao izlaz odgovarajući program na mašinskom jeziku.

Korišćenjem jezika višeg nivoa opis naredbi i podataka vrši se na način blizak prirodnom (engleskom) jeziku. U ovim jezicima jednoj naredbi odgovara više naredbi simboličkog jezika. Važno je napomenuti da ovi jezici imaju visok stepen nezavisnosti u odnosu na arhitekturu računara i operativni sistem na kojem se izvršavaju. S obzirom na to da računar razume samo program napisan na mašinskom jeziku, svaki program pisan jezikom višeg nivoa mora se prevesti na mašinski jezik.

Ne postoji striktna definicija programskih jezika, neformalno ali vrlo intuitivno možemo reći da je programski jezik jezik konstruisan formalno da bi se omogućilo zadavanje instrukcija mašinama, posebno računarima.

- Programski jezik[3] je veštački jezik za opis konstrukcija (pisanje instrukcija) koje mogu biti prevedene u mašinski jezik i izvršene od strane računara. (American Heritage Dictionary)
- Programski jezik[2] predstavlja notaciju (sistem označavanja) namenjenu izražavanju programa na način koji može prihvatiti računar kojimse opisuje računarski postupak tj. zapisivanju algoritamskog rešenja zadatka. To je skup osnovnih elemenata jezika-leksema (leksičkih simbola, završnih simbola) sa definisanim pravilima građenja pravilnih konstrukcija programskog jezika od tih elemenata.

Danas[8] postoji mnogo programskih jezika. U jednom smislu su oni svi jednaki, jer korišćenjem bilo kog jezika može se isprogramirati manje više sve što želite, na primer može se kreirati veb stranica koristeći programski jezik Ruby, Java, Python, C# ili JavaScript. Osnova ove činjenice je da svi programski jezici služe istoj svrsi, i oni su na najosnovnijem nivou isti. Međutim jezici često prave kompromise u pogledu pogodnosti, sigurnosti i brzine, i to je razlog zbog koga se određeni jezici koriste za određenu vrstu poslova. Upravo je postojanje raznovrsnih poslova koje treba obaviti razlog postojanja velikog broja programskih jezika, a kompromis je diktiran tim poslom.

Enciklopedija Britanika pominje preko 2000 programskih jezika, drugi izvori pominju preko 2500 dokumentovanih programskih jezika, dok Diarmuid Pigott sa Univerziteta iz Australije u svom radu „Encyclopedia of Computer Languages” navodi preko 8000 jezika.

Na slici 1 prikazani su programski jezici, svakako ne svi, ali jesu oni koji su najčešće u upotrebi.

- Funkcionalna paradigma – Jezici funkcionalne paradigme posmatraju izračunavanje kao process izračunavanja matematičkih funkcija i izbegavaju koncept stanja i promenljive. Koreni funkcionalnog programiranja leže u lambda računu razvijenom 1930-ih kako bi se izračunao process definisanja i primene matematičkih funkcija i rekurzija. Mnogi funkcionalni jezici se smatraju nadogradnjom lambda-računa. Najznačajniji jezici su Lisp, Scheme, ML, Haskell, ..
- Logička paradigma – Logički jezici su po definiciji jezici koji koriste matematičku logiku za programiranje računara.
- Objektno-orijentisana paradigma – Jezici objektno-orijentisane paradigme koriste objekte – specijalizovane strukture podataka koje uz polja podataka sadrže i metode kojima se manipuliše podacima. Podaci se mogu obrađivati isključivo primenom metoda što smanjuje zavisnot između različitih komponenta programskog koda i čini ovu paradigmu pogodnu za razvoj velikih aplikacija uz mogućnost saradnje većeg broja programera. Najznačajniji jezici su C++, Java, Eiffel, Simula. . .

Definiciju programskih jezika možemo prilagoditi osnovnim programskim paradigmama.

“Programski jezik[1] je sredstvo koje koristi čovek da izrazi *proces* pomoću kojeg računar rešava nekakav *problem*. “

U zavisnosti od toga na kojoj od ovih reči je akcenat, programskim jezikom je podržana dominantna programska paradigma:

- Čovek – logička paradigma
- Process – funkcionalna paradigma
- Računar – proceduralna paradigma
- Problem – objektno-orijentisana paradigma

Programske paradigme su usko povezane sa programskim jezicima. Rešavanje problema zahteva izbor odgovarajućeg koncepta, ali čak i mali problem zahteva različit skup koncepta i metoda za rešavanje. Upravo iz tog razloga programski jezici bi trebalo da podržavaju više paradigmi.

4 Zaključak

Izrada programskih jezika koji su lako čitljivi i razumljivi je od ključnog značaja. Zbog čitljivosti koda lakše je posle izvesnog vremena modifikovati izvorni kod, dodati nove funkcije...što je jedan od ključnih razloga nastanka programskih jezika. Programeri su želeli da kod bude čitljiv i lakši za rukovanje. Takođe, sve bržim razvojem računarstva i informatike ljudi širom sveta su počeli da koriste računare za različite poslove pa su se tako i programski jezici razvijali u pravcu poslova za koje su namenjeni, ili nastajali novi.

Kaže se da čovek vredi onoliko koliko jezika govori. Svakim jezikom koji naučite, postajete bogatiji i osećate se bolje. A šta se dešava kada se taj odnos primeni na programere i njihove programske jezike? Zapravo, važi ista deviza – programeri koji se služe različitim vidovima koda su visoko cenjeni.

Literatura

- [1] Dizajn programskih jezika. http://www.programskijezici.matf.bg.ac.rs/DizajnProgramskihJezika.html#0_tab. Accessed: 2018-11-11.
- [2] E-škola. <https://profsrboljubmilosavljevic.wordpress.com>. Accessed: 2018-10-25.
- [3] American Heritage Publishing Company. *The American heritage dictionary of the English language*. Houghton Mifflin, Boston, 4th edition, 2006.
- [4] William Findlay David A. Watt and John Hudges. *Programming Language Concepts Paradigms*. Prentice Hall Ptr, Boston, 1th edition, 1993.
- [5] Milan Šipka Ivan Klajn. *Veliki rečnik stranih reči i izraza -kutija*. Prometej, Beograd, 4th edition, 2011.
- [6] F.Marić P.Janičić. *Programiranje 2: osnove programiranja kroz jezik C*. Matematički fakultet Univerziteta u Beogradu, Beograd, 1th edition, 2015.
- [7] Allen B. Tucker and Robert E. Noonan. *Programming Languages: Principles and Paradigms*. McGraw-Hill, Boston, second edition, 2007.
- [8] Wallace Wang. *Beginning Programming For Dummies*. Wiley Publishing, Inc., 111 River Street, Hoboken, NJ 07030-5774, 3th edition, 2004.