

Sąlygos sakiny

Programos atlieka veiksmus su duomenimis (pvz.: sveikaisiais, realiaisiais skaičiais; simboliais; simbolių eilutėmis). Duomenys gali būti pastovūs (pvz., gimimo data) ir kintami (pvz., amžius). Duomenys, kurie atliekant programą nekinta, vadinami konstantomis.

Kintamieji skirti duomenims, kurių reikšmės keičiasi atliekant programą, atmintyje laikyti. Jų vardai sudaromi iš raidžių ir skaitmenų, tačiau pirmas ženklas turi būti raidė. Parenkant kintamųjų vardus reikia stengtis, kad jie pažymėtų duomenų paskirtį. Pavyzdžiui, jei skaičiuojame trijų skaičių sumą, tai juos atitinkančius kintamuosius patogų žymėti a, b, c, o sumą – vardu suma. Kintamieji gali būti aprašomi bet kurioje programos vietoje, bet prieš juos panaudojant veiksmuose. Dažniausiai jie aprašomi funkcijos main() pradžioje. Kiekvienam kintamajam reikia nurodyti, kokio tipo duomenis jis turi laikyti atmintyje. Duomenų tipas apibrėžia tam tikrą aibę reikšmių ir veiksmų, kuriuos galima atlikti su tomis reikšmėmis.

intamojo galiojimo pradžia – jo aprašymo vieta. Kintamasis galioja tame programos bloke, kuriame jis yra apibrėžtas. Bloką sudaro programavimo kalbos sakinių seka, parašyta tarp riestinių skliaustų {}. Jis skiriasi nuo sudėtinio sakinio tuo, kad jame yra ir sakiniai, ir kintamųjų aprašai. Dažniausiai blokas naudojamas funkcijoms užrašyti.

Sąlyginių sakinių programoje keičiama nuosekli sakinių atlikimo tvarka: jei Sąlyga tenkinama, atliekamas PirmasSakiny, jei ne – po else esantis AntrasSakiny. Jeigu reikia atlikti kelis sakinius, kai Sąlyga tenkinama arba netenkinama, tai jie rašomi tarp { ir }. Sąlyga – bet koks santykio arba loginis reiškiny.

Sudėtinis sakiny – tai tarp riestinių skliaustų {} parašyta sakinių seka.

Aprašant kintamuosius, reikia prisiminti: Kintamieji, aprašyti prieš pagrindinę funkciją main(), vadinami globaliaisiais. Jie galioja visoje programoje. Kintamieji, aprašyti funkcijoje, vadinami lokaliaisiais. Jie galioja tik toje funkcijoje, išėjus už funkcijos ribų, šių kintamųjų reikšmės neišsaugomos. Jei kintamasis tokiu pačiu vardu aprašytas prieš pagrindinę funkciją main() ir funkcijoje, pirmenybė suteikiama lokaliajam, t. y. funkcijoje globalusis kintamasis negalioja.

Priskyrimo sakiny naudojamas, kai kintamajam reikia suteikti reikšmę programos tekste. Priskyrimo sakinio struktūra tokia: kintamojoVardas = Reiškiny;

Čia simbolis = žymi priskyrimo operaciją, o Reiškiny nurodo, kokius veiksmus, kokia tvarka ir su kokiais argumentais reikia atlikti. Kairiojoje priskyrimo operacijos pusėje įrašytas kintamojo vardas nurodo, kam atmintyje suteikiama apskaičiuota reiškinio reikšmė. Pavyzdžiui, priskyrimo sakiny $y = x * x$; reiškia, kad argumento x reikšmė kelinama kvadratu ir rezultatas priskiriamas kintamajam y. Būtina, kad priskyrimo operacijos kairėje pusėje nurodyto kintamojo tipas atitiktų reiškinio, esančio dešiniojoje pusėje, reikšmės tipą. Reiškiniuose galima naudoti tik tokius kintamuosius, kurių reikšmės apibrėžtos anksčiau įrašytuose programos sakiniuose.

Kalboje C++ įvedimo / išvedimo sistema yra susijusi su srauto sąvoka. Įvedimo / išvedimo srautas – tai loginis įrenginy, kuris išveda ir priima naudotojo informaciją. Įvedimo / išvedimo sistema susieja srautą su fiziniu įrenginiu (klaviatūra, ekranu, diskų įtaisais). Visi įvedimo / išvedimo srautai veikia vienodai nepriklausomai nuo fizinių įrenginių charakteristikų.

Duomenims įvesti iš standartinio įvedimo srauto cin, paprastai susieto su klaviatūra, ir / arba iš srauto, susieto su failu, dažniausiai naudojamas operatorius >>. Asmeniniuose kompiuteriuose standartinis įvedimo įtaisas yra klaviatūra.

Kai pradinių duomenų daug, juos įvedinėti klaviatūra nėra patogiu. Be to, norint skaičiavimus pakartoti, reikia duomenis įvesti iš naujo. Nepatogumų galima išvengti, duomenis iš anksto įrašant į tekstinį failą. Tačiau į failą nerašomi pranešimai ir paaiškinimai, kokie duomenys ir kokia eilės tvarka pateikiami. Rašant duomenų skaičiavimo iš failo sakinius, reikia iš anksto žinoti, kokia eilės tvarka faile surašyti duomenys. Nuo to priklauso, kiek bus kintamųjų ir kokia eilės tvarka jų reikšmės bus skaitomos iš failo.