

C++ TILIDA KONSTRUKTOR VA DESTRUKTOR

КОНСТРУКТОР И ДЕСТРУКТОР В С++

CONSTRUCTOR AND DESTRUCTOR IN C++

Andijon davlat universiteti

Axborot texnologiyalari va kompyuter injineriingi fakulteti

Informatika o'qitish metodikasi yo'nalishi mалабаси

Xonkeldieva Feruzaxon Nodirovna

Annotasiya Maqolada har qanday dasturning strukturasi tarmoqlanishi, sikllar to'plamining kombinasiyasi, obektga mo'ljallangan yondoshuv tog'risida, ob'jekt - mavhum (abstrakt) mohiyat bo'lib, u bizni o'rab turgan haqiqiy olamning tavsiflariga ega ekanligi keng yoritib berildi. Obektga mo'ljallangan yondoshuv yeng muvaffaqiyatli, vaqt sinovidan o'tgan uslublarni o'zida samarali mujassam etadi.

Аннотация В статье структура любой программы — это разветвление, сочетание набора циклов, объектно-ориентированного подхода и того, что объект — это абстрактная сущность, имеющая описания реального мира, который нас окружает. Оно было широко распространено. Объектно-ориентированный подход эффективно включает в себя успешные, проверенные временем методы.

Annotation In the article, the structure of any program is branching, a combination of a set of loops, an object-oriented approach, and the fact that an object is an abstract essence that has characteristics of the real world that surrounds us. It was widely distributed. An object-oriented approach effectively incorporates successful, time-tested techniques.

Kalit so'zlar: dastur, ob'jekt, C++tili, atribut, konstruktor, destruktur, polimorfizm.

Ключевые слова: программа, объект, язык C++, атрибут, конструктор, деструктор, полиморфизм.

Keywords: program, object, C++ language, attribute, constructor, destructor, polymorphism.

Har qanday dasturning strukturasi tarmoqlanish va sikllar to'plamining kombinasiyasidan iborat bo'ladi. Ob'ektga mo'ljallangan yondoshuv bir kunda o'ylab topilgan emas. Uning paydo bo'lishi dasturiy ta'minotning tabiiy rivojidadagi navbatdagi



pog'ona xolos. Vaqt o'tishi bilan qaysi uslublar ishlash uchun qulay qaysinisi noqulay yekanini aniqlash oson bo'lib bordi. Obektga mo'ljallangan yondoshuv eng muvaffaqiyatli, vaqt sinovidan o'tgan uslublarni o'zida samarali mujassam yetadi. Ob'yekt mavhum (abstrakt) mohiyat bo'lib, u bizni o'rab turgan haqiqiy olamning tavsiflariga ega.

Ob'yektlarni yaratish va ular ustida manipulyatsiyalar olib borish C++tilining qandaydir alohida imtiyozi emas, balki ob'yektlarning tavsifi va ular ustida o'tkaziladigan operatsiyalarni kodli konstruktsiyalarda o'zida mujassamlantiradigan dasturlash metodologiyasi (uslubiyoti) ning natijasidir. Dasturning har bir ob'yekti, har qanday haqiqiy ob'yekt kabi, o'z atributlari va o'ziga xos xulq-atvori bilan ajralib turadi. Ob'yektlarni turli kateqoriyalarga ko'ra tasniflash mumkin- masalan, myening «Cassio» raqamli qo'l soatim soatlar sinfiga mansub. Soatlarning dasturiy ryealizatsiyasi (ishlatilishi), standart holat sifatida, sizning kompyutyeringizning operatsiya tizimi tarkibiga kiradi [1].

C++ da hamma narsa atributlari va usullari bilan bir qatorda sinflar va ob'ektlar bilan bog'liq. Masalan, real hayotda mashina ob'ektdir. Mashinada og'irlik va rang kabi atributlar, haydash va tormoz kabi usullar mavjud. Xususiyatlar va usullar asosan sinfga tegishli o'zgaruvchilar va funktsiyalardir. Ular ko'pincha "sinf a'zolari" deb nomlanadi. Sinf - bu bizning dasturimizda foydalanishimiz mumkin bo'lgan foydalanuvchi tomonidan aniqlangan ma'lumotlar turi va u ob'ektlar konstruktori yoki ob'ektlarni yaratish uchun "loyiha" sifatida ishlaydi.

Ob'yektni yaratishda uni inisializatsiyalash kerak. Bu maqsadda C++ tilida konstruktor deb nomlanuvchi maxsus funksiya-a'zo aniqlangan. Sinf konstruktori har safar sinf ob'yekti yaratilishi paytida chaqiriladi. Konstruktor nomi o'zi a'zo bo'lgan sinf nomi bilan ustma-ust tushadi va qaytaruvchi qiymatga ega bo'lmaydi. "Konstruktor ishladi \n"; } //... Keltirilgan dastur bo'lagida konstruktor bajaradigan ish oldingi misoldagi konstruktor ishi bilan ekvivalent. Albatta, sinf elementlarini inisializasiya qilishning qaysi shaklini qo'llash dastur tuzuvchiga bog'liq. Biroq, shunday holatlar bo'ladiki, unda elementlarni inisializatsiyalash ro'yxatidan foydalanmaslikning iloji yo'q- sinfning berilganlar-konstantalariga va ko'rsatgichlariga boshlang'ich qiymat berishda, sinf a'zosi ob'yekt



bo'lganda va bu ob'yekt konstruktori bir yoki bir nechta parametrlarga qiymat berishni talab qilgan hollarda [2].

konstruktornlarni ishlatishda kolliziyadan (bir narsani ikki xil tushunishdan) qochish kerak, ya'ni sinfda bir nechta konstruktor bo'lganda kompilyator qachon ularning qaysi biri chaqirilishini aniq bilishi kerak. Quyidagi misolda bu holat bilan bog'liq xato ko'rsatilgan.

```
class S { public: S(), // Kelishuv bo'yicha konstruktor S(int i=0); // Kelishuv bo'yicha konstruktor o'rnida //ishlatilishi mumkin bo'lgan konstruktor }, int main() { S ob1(10), // S::S(int) konstruktori ishlatiladi. S ob2, //Noto'g'ri. S::S(int) yoki S::S() // konstruktornlarning qaysi biri // chaqirilishi noma'lum. Bu ziddiyatni hal qilish yo'li – bu sinf e'lonidan kelishuv bo'yicha konstruktorni o'chirishdir.
```

Konstruktor va destruktord - bu o'z sinflari bilan bir xil nomdagi a'zolar funktsiyalari. Avvalgi turi konstruktor ob'ektzni ishga tushirishda yordam beradi. Aksincha, a halokatchi hech qanday foydasi bo'lmaganda yaratilgan konstruktorni o'chirib tashlaydigan konstruktordan farq qiladi.

Ba'zan foydalanishdan oldin ob'ektning bir qismini boshlash kerak bo'ladi. Masalan, biz stek ustida ishlayapmiz, biron bir amalni bajarishdan oldin stakning yuqori qismi har doim nolga o'rnatilishi kerak. Avtomatik ishga tushirishning ushbu xususiyati "Konstruktor" orqali amalga oshiriladi. Shunga o'xshab, agar ob'ekt yo'q qilinishidan oldin ba'zi kodlarni bajarishi kerak bo'lsa. Masalan, ob'ekt yo'q qilinishidan oldin ochilgan faylni yopishi kerak bo'lsa. Bu "Destructor" yordamida amalga oshirilishi mumkin. Keling, taqqoslash jadvali yordamida konstruktor va destruktord o'rtasidagi ba'zi asosiy farqlarni ko'rib chiqamiz [3].

Konstruktorning ta'rif A konstruktor asosan sinfning a'zosi funktsiyasidir, u ob'ektzni ishga tushiradi va unga xotira ajratadi. Konstruktorlarni osongina aniqlash mumkin, chunki ular e'lon qilingan va sinf bilan bir xil nom bilan aniqlangan. Konstruktord qaytish turi mavjud emas; shunday qilib, ular hech narsa qaytarmaydi, hatto "bekor" ham emas. Konstruktor har doim sinfning ochiq qismida aniqlanadi.

Sinfda bir nechta konstruktor bo'lishi mumkin; ularni o'tgan argumentlar soni va turiga qarab ajratish mumkin. Agar sinfda bir nechta konstruktor mavjud bo'lsa, yopiq konstruktor

(hech narsa qilmaydigan konstruktor) ular bilan birga aniqlanishi kerak, u hech narsa qilmaydi, lekin kompilyatorni to'ldiradi. Konstruktorlarni standart argumentlar bilan ham aniqlash mumkin. Holbuki, ular ob'ektni "dinamik ravishda" ishga tushiradilar. Konstruktorlar na meros qilib olinishi mumkin, na virtual, balki ular haddan tashqari yuklanishi mumkin. Ularni manziliga yo'naltirish mumkin emas.

Destruktorning ta'rifi A Yo'q qiluvchi shuningdek, ob'ektga ajratilgan xotirani taqsimlaydigan sinfning a'zosi funktsiyasi. U sinf nomi bilan bir xil nom bilan belgilanadi, oldin a tilda (~) belgi. Destruktorlar har doim konstruktorlarning teskari tartibida chaqiriladi [4].

Haddan tashqari yuklangan qavslar Tegishli munozarani topishingiz mumkin Muloqot-C ++ sinflari, iltimos, bahsli bayonotlar mavjudligini ta'minlashga yordam bering ishonchli manbalar. (2009 yil yanvar) (Ushbu shablon xabarini qanday va qachon olib tashlashni bilib oling) Kvadrat qavs va dumaloq qavs C ++ tuzilmalarida haddan tashqari yuklanishi mumkin. Kvadrat qavs to'liq bitta argumentni o'z ichiga olishi kerak, dumaloq qavs esa istalgan aniq sonli argumentni o'z ichiga olishi mumkin yoki argumentsiz.

Yuqorida ko'rsatilgan operatorlardan tashqari o'q operatori (->), yulduzcha ko'rsatilgan o'q (->*), the yangi kalit so'z va o'chirish kalit so'zni ortiqcha yuklash ham mumkin. Ushbu xotira yoki ko'rsatgich bilan bog'liq operatorlar haddan tashqari yuklanganidan keyin xotira ajratish funktsiyalarini qayta ishlashlari kerak. Topshiriq singari (=) operatori, agar ular aniq bir deklaratsiya qilinmasa, ular sukut bo'yicha ortiqcha yuklanadi.

Inkapsulyatsiya. Agarda muxandis ishlab chiqarish jarayonida rezistorni qo'llasa, u buni yangidan ixtiro qilmaydi, omborga (magazinga) borib mos parametrlarga muvofiq kerakli detalni tanlaydi. Bu qolda muqandis joriy rezistor qanday tuzilganligiga e'tiborini qaratmaydi, rezistor faqatgina zavod xarakteristikalariga muvofiq ishlasa etarlidir. Aynan shu tashqi konstruktsiyada qo'llaniladigan yashirinlik yoki ob'ektni yashirinligi yoki avtonomligi xossasi inkapsulyatsiya deyiladi.

Merosxo'rlik Acme Motors kompaniyasi injenerlari yangi avtomobil konstruktsiyasini yaratishga aqd qilishsa, ular ikkita variantdan birini tanlashlari lozim. Birinchisi, avtomobilning konstruktsiyasini boshidan boshlab yangidan ixtiro qilish, ikkinchisi esa



mavjud Star modelini o'zgartirishdir. Star modeli qariyb ideal, faqatgina unga turbokompressor va olti tezlanishli uzatma qo'shish lozim. Bosh muqandisikkinchi variantni tanladi. Ya'ni noldan boshlab qurishni emas, balki Star avtomobiliga ozgina o'zgartirish qilish orqali yaratishni tanladi. Uni yangi imkoniyatlar bilan rivojlantirmoqchi bo'ldi. Shuning uchun, yangi modelni Quasar deb nomlashni taklif qildi. Quasar-Star modeliga yangi detallarni qo'shish orqali yaratilgan. C++ tili xam shunday merosxo'rlikni qimoya qiladi. Bu yangi berilganlar tipi (sinf), oldindan mavjud bo'lgan sinfni kengaytirishdan qosil bo'ladi. Bunda yangi sinf oldingi sinfning merosxo'ri deb ataladi [5].

Odatiy konstruktor Bu konstruktorga hech qanday dalil berilmagan konstruktor. Odatiy konstruktorlarda parametr yo'q, lekin sukut bo'yicha konstruktorga qiymatlar sukut bo'yicha (dinamik ravishda) o'tkazilishi mumkin.

Parametrlangan konstruktor Ushbu turdagi konstruktor argumentlarni qabul qiladi, ma'lumotlar a'zolariga turli xil qiymatlarni argument sifatida berishimiz mumkin.

Konstruktorni nusxalash Nusxalashtiruvchi konstruktor boshqa turdagi konstruktorlardan farq qiladi, chunki u boshqa ob'ekt manzilini argument sifatida qabul qiladi [6].

Konstruktorlar ko'pincha deklaratsiya bilan bir xil nomga ega sinf. Ularning vazifasi bor boshlash ob'ektniki ma'lumotlar a'zolari va tashkil etish sinfning o'zgarmasligi, agar o'zgarmas narsa yaroqsiz bo'lsa. To'g'ri yozilgan konstruktor natijani qoldiradi ob'ekt yaroqli davlat. O'zgarmas narsalar konstruktorlarda initsializatsiya qilinishi kerak [7].

C++ tili paydo bo'lgandan beri, o'qitish, ilmiy izlanish, tijorat dasturiy ta'minotlarini yaratish bo'yicha amaliy til sifatida tobora ko'proq e'tiborga ega bo'lmoqda. Mazkur davr mobaynida C++ tili yangi imkoniyatlar bilan boyib bormoqda, ular yangi kiritilgan funksiyalar (shablonlar va istisnolarga ishlov berish) hisobiga amalga oshirildi. Dasturlashni yangi o'rganayotganlarning oldida ko'pincha- Albatta, dasturlash kabi murakkab jarayonni ko'rsatmalar ko'rinishidagi qo'llanma sifatida rasmiylashtirish mumkin emas. Shunga qaramay, dasturlashni qadammaqadam tasvirlovchi qo'llanma hamma vaqt ham foydali bo'ladi. Albatta, talabalar turli dasturlarni yoza olish imkoniyatiga ega bo'lishi kerak, ammo bunda ular yaxshi ko'rsatgichlarga ega bo'la olishlari bo'yicha ishonch zarur bo'ladi. C++



tiliga kirish sifatida talqin etilishi mumkin. Ushbu maqolada C++ tilida dasturlash bilan birgalikda, o'z imkoniyatlaridan to'g'ri foydalanishni hamda ob'jektga yo'naltirilgan dasturlash asoslarini ham kengroq yoritib bera oladi. Ushbu maqolada xar bir qismida ma'lumotlar qisqa va lo'nda holda keltirilishiga harakat qilingan. Shuni qayd etish lozimki, keltirilgan misollar, ushbu kurs ishida talqin etilgan nazariy bilimlarni amaliyotda qo'llash imkoniyatini yaratadi.

FOYDALANILGAN ADABIYOTLAR RO'YXATI

1. Karimov I.A. Xavfsizlik va barqarorlik yo'lida. 6-jild. Toshkent "O'zb ekiston". 1998. 409 b.
2. Horstman C.S. C++ for Everyone, 2 edition-2011, 562 p
3. Herb Sutter. More Exceptional C++. 2007- 304 p.
4. Nazirov Sh.A., Qobulov R.V., Bobojanov M.R., Raxmanov Q.S. C va C++ tili. "Vorish-nashriyot" MCHJ, Toshkent 2013, 488 b
5. Aripov M., Begalov B., Begimqulov U., Mamarajabov M. Axborot texnologiyalari. Toshkent: Noshir, 2009. -368 s.
6. Deitel P.J., Deitel H.M. C++. How to Program, 9 th Edition-2011.-1070
7. Подбелский В., Фомин С. Программирование на языке Си. Учебное пособие. -2-издание.- М.: Финансы и статистика, 2004. -600 с.