

C++ TILIDA SWITCH-CASE OPERATORIDAN FOYDALANISH

ИСПОЛЬЗОВАНИЕ ОПЕРАТОРА SWITCH-CASE В C++

USING THE SWITCH-CASE OPERATOR IN C++

Shermatova Xilola Mirzayevna

FarDU Axborot texnologiyalari kafedراسي dotsenti

shermatovahilola1978@gmail.com

<https://orcid.org/0000-0001-5014-9549> +998904099095

Ibrohimjonov Shoxzod Suhbatjon o'g'li

FarDU Axborot tizimlari va texnologiyalari yo'nalishi 1-kurs talabasi

Shohzod.ibrohimov@gmail.com

Annotatsiya: *Ushbu maqolada C++ dasturlash tilidagi switch-case operatorining ishlash prinsipi va uning dasturlashdagi ahamiyati haqida so'z boradi. switch-case operatori, ko'plab shartlarni tekshirishda samarali va qisqa kod yozishga yordam beradi. Maqolada operatorning sintaksisi va ishlash tartibi keltirilgan, shuningdek, amaliy misollar yordamida uning qanday ishlatilishi tushuntirilgan. switch-case operatorining afzalliklari, xususan, katta miqdordagi shartlarni tekshirishda kodning o'qilishi va samaradorligini oshirishga qanday yordam berishi haqida ham muhokama qilinadi. Ushbu maqola C++ dasturlash tilini o'rganayotgan dasturchilar uchun foydali bo'lishi mumkin.*

Kalit so'zlar: C++, switch-case, operator, dasturlash, shart tekshirish, kod samaradorligi, if-else, kod bloklari, sintaksis, amaliy misol, dasturchilar.

Аннотация; *В данной статье рассматривается оператор switch-case в языке программирования C++, его принцип работы и значимость в программировании. Оператор switch-case помогает писать более эффективный и компактный код при проверке нескольких условий. В статье приводятся синтаксис и принцип работы оператора, а также практические примеры его использования. Также обсуждаются преимущества оператора switch-case, в частности, как он улучшает читаемость и эффективность кода при проверке множества условий. Эта статья будет полезна программистам, изучающим C++.*

Ключевые слова: *C++, switch-case, operator, programming, condition checking, code efficiency, if-else, code blocks, syntax, practical example, developers.*

Abstract: *This article discusses the switch-case operator in C++ programming language, explaining its functionality and significance in programming. The switch-case operator helps write more efficient and concise code when checking multiple conditions. The article covers the syntax and working principle of the operator, with practical examples illustrating its use. Additionally, it discusses the advantages of the switch-case operator, particularly how it improves code readability and efficiency when checking numerous conditions. This article can be useful for programmers learning C++.*

Keywords: *C++, switch-case, operator, programming, condition checking, code efficiency, if-else, code blocks, syntax, practical example, developers.*

Kirish; Zamonaviy dasturlash tillarida dastur oqimini boshqarish vositalari muhim o‘rin egallaydi. Ayniqsa, turli shartlarga qarab qaror qabul qilish talab etiladigan vaziyatlarda shartli operatorlar dastur logikasining asosi hisoblanadi. C++ dasturlash tilida bunday operatorlardan biri — switch-case operatoridir. Bu operator o‘zgaruvchining qiymatiga qarab, tegishli kod blokini bajarish imkonini beradi va shartlar soni ko‘p bo‘lgan holatlarda kodni soddalashtirishda, o‘qilishini yaxshilashda katta yordam beradi.

Odatda, dasturchilar ko‘p hollarda if-else konstruksiyasidan foydalanadilar. Biroq, if-else tuzilmasi murakkab bo‘lgan holatlarda, ya’ni shartlar soni ortgan sari, kod nafaqat uzunlashadi, balki o‘qilishi va tushunilishi ham qiyinlashadi. Shu nuqtai nazardan switch-case operatori — kodni optimallashtirishning qulay vositalaridan biridir. U nafaqat ko‘rinish jihatidan ixcham, balki samaradorlik nuqtai nazaridan ham yaxshi natija beradi, ayniqsa, diskret qiymatlar ustida ishlayotgan dasturlarda.

Ushbu maqolada C++ tilida switch-case operatorining sintaksisi, ishlash prinsipi, va amaliy dasturlarda qanday ishlatilishi batafsil ko‘rib chiqiladi. Maqolada real hayotdagi vaziyatlarga mos misollar yordamida operatorning afzalliklari va cheklovlari haqida ham fikr yuritiladi. Maqola C++ dasturlash tilini o‘rganayotgan boshlang‘ich va o‘rta darajadagi dasturchilar uchun foydali bo‘lishi mumkin. Shuningdek, operatorning samarali ishlatilishi orqali kod sifati, tushunarligi va texnik xizmat ko‘rsatish qulayligini oshirish yo‘llari ham yoritiladi.

Dasturning kodi :

```
#include <iostream>
using namespace std;
int main() {
double son1, son2, natija;
char amal;
cout << "1-sonni kiriting: ";
cin >> son1;
cout << "Amalni tanlang (+, -, *, /): ";
cin >> amal;
cout << "2-sonni kiriting: ";
cin >> son2;
switch (amal) {
case '+':
natija = son1 + son2;
cout << "Natija: " << natija << endl;
break;
case '-':
natija = son1 - son2;
cout << "Natija: " << natija << endl;
break;
case '*':
natija = son1 * son2;
cout << "Natija: " << natija << endl;
break;
case '/':
if (son2 != 0) {
natija = son1 / son2;
cout << "Natija: " << natija << endl;
} else {
cout << "Nolga bo'lish mumkin emas!" << endl;
return 1;
}
break;
default:
cout << "Noto'g'ri amal tanlandi!" << endl;
return 1;
}
```

```
return 0;
```

```
}
```

Dastur tahlili:

#include <iostream> – iostream kutubxonasi C++ **standarti** bo‘lib, ekranga chiqish (cout) va foydalanuvchidan kiritish (cin) uchun ishlatiladi.

using namespace std; – std::cout, std::cin deb yozmaslik uchun kerak. Agar bu bo‘lmasa, har bir chiqish va kiritish operatori oldidan **std::** yozish kerak bo‘ladi.

int main() – Har qanday C++ dasturining boshlang‘ich nuqtasi. Dastur shu yerdan ishga tushadi.

Double son1, son2; – Haqiqiy sonlar uchun ikkita o‘zgaruvchi (chunki natija butun son bo‘lishi shart emas).

char amal; – Matematik amalni (+, -, *, /) saqlash uchun **char** tipidagi o‘zgaruvchi.

cin >> son1; – Foydalanuvchidan birinchi sonni kiritishni so‘raydi va son1 o‘zgaruvchisiga saqlaydi.

cout – "Amalni tanlang (+, -, *, /): "; degan yozuvni chiqarib beradi.

cin >> amal; – Foydalanuvchi **matematik amalni** tanlaydi (+, -, *, /) va **amal** o‘zgaruvchisiga saqlanadi.

cout – "2-sonni kiriting: " degan xabarni chiqaradi.

cin >> son2; – Foydalanuvchidan ikkinchi sonni kiritishni so‘raydi va son2 o‘zgaruvchisiga saqlaydi.

natija – Hisoblangan qiymatni saqlash uchun o‘zgaruvchi.

switch (amal) – amal o‘zgaruvchisini tekshiradi va unga mos keladigan operatsiyani bajaradi.

Qo‘shish amali:

Agar amal + bo‘lsa, son1 va son2 yig‘iladi.

break; – switch dan chiqib ketish uchun.

Ayirish amali:

Agar amal - bo‘lsa, son1 dan son2 ayiriladi.

Ko‘paytirish amali:

Agar amal * bo‘lsa, son1 va son2 ko‘paytiriladi.

Bo‘lish amali va 0 ga bo‘lishni tekshirish

Agar amal / bo‘lsa, bo‘lish amali bajariladi.

if (son2 != 0) – Nolga bo‘lish mumkin emas, shuning uchun avval son2 tekshiriladi.

Agar son2 == 0 bo‘lsa, "Nolga bo‘lish mumkin emas!" xabari chiqib, dastur xato bilan tugaydi (return 1;).

Agar foydalanuvchi +, -, *, / dan boshqa amal kiritsa, "Noto'g'ri amal tanlandi!" xabari chiqib, dastur **xato bilan tugaydi** (return 1;).

cout << "Natija: " << natija << endl; – Hisoblangan natija ekranga chiqariladi.

return 0; – dastur to'g'ri ishlaganligini bildiradi (0 qaytaradi).

Dastur natijasi:

1. Ma'lumot kiritish

Dastur **foydalanuvchidan birinchi sonni** kiritishni so'raydi.

Keyin **qaysi amalni bajarishni** tanlashni so'raydi (+, -, *, /).

So'ng, **ikkinchi son** kiritiladi.

2. Amalni tekshirish va bajarish

Dastur switch operatori yordamida **amal turini tekshiradi:**

+ bo'lsa – **qo'shish**

- bo'lsa – **ayirish**

* bo'lsa – **ko'paytirish**

/ bo'lsa – **bo'lish**

Agar foydalanuvchi **noto'g'ri amal kiritgan bo'lsa**, dastur **xato xabari** chiqaradi.

3. Nolga bo'lishni tekshirish

Agar foydalanuvchi **nolga bo'lishga** harakat qilsa, dastur **xatolik haqida xabar** beradi va hisoblashni to'xtatadi.

Natijani chiqarish

Hisoblangan natija ekranga chiqariladi.

Dastur bajarilishini yakunlaydi.

Xulosa: C++ dasturlash tilida switch-case operatori shartli operatorlarning samarali va ixcham variantlaridan biridir. U, ayniqsa, bir nechta variantlar orasidan tanlov qilish talab qilinadigan vaziyatlarda kodni soddalashtirishda va samaradorligini oshirishda katta yordam beradi. if-else konstruktsiyasiga nisbatan, switch-case operatori kodning o'qilishi va tushunilishini osonlashtiradi, ayniqsa ko'p shartlar mavjud bo'lganda.

Maqolada keltirilgan kalkulyator dasturi misolida switch-case operatorining sintaksisi, ishlash prinsipi va amaliy dasturlarda qanday qo'llanilishi batafsil ko'rsatib o'tildi. switch operatori har bir shartga alohida holatlarni (case) taqdim etish imkonini beradi, va default holati orqali noto'g'ri kiritilgan qiymatlar uchun xatoliklarni boshqarish mumkin.

Umuman olganda, switch-case operatori kodning samarali va o'qilishi oson bo'lishini ta'minlab, dasturchilarga murakkab shartli tarmoqlanishlarni

soddalashtirishda yordam beradi. Bunday xususiyatlari bilan switch-case C++ dasturlash tilida keng qoʻllaniladigan vositadir va dasturchilarning samaradorligini oshiradi.

FOYDALANILGAN ADABIYOTLAR:

1. Stroustrup, B. (2013). The C++ Programming Language (4th Edition). Addison-Wesley.
2. C++ Documentation (2025). C++ Standard Library - Switch Statement [online].
3. Deitel, H. M., & Deitel, P. J. (2018). C++ How to Program (10th Edition). Pearson.
4. Bjarne Stroustrup. (2020). Programming: Principles and Practice Using C++ (2nd Edition). Addison-Wesley.