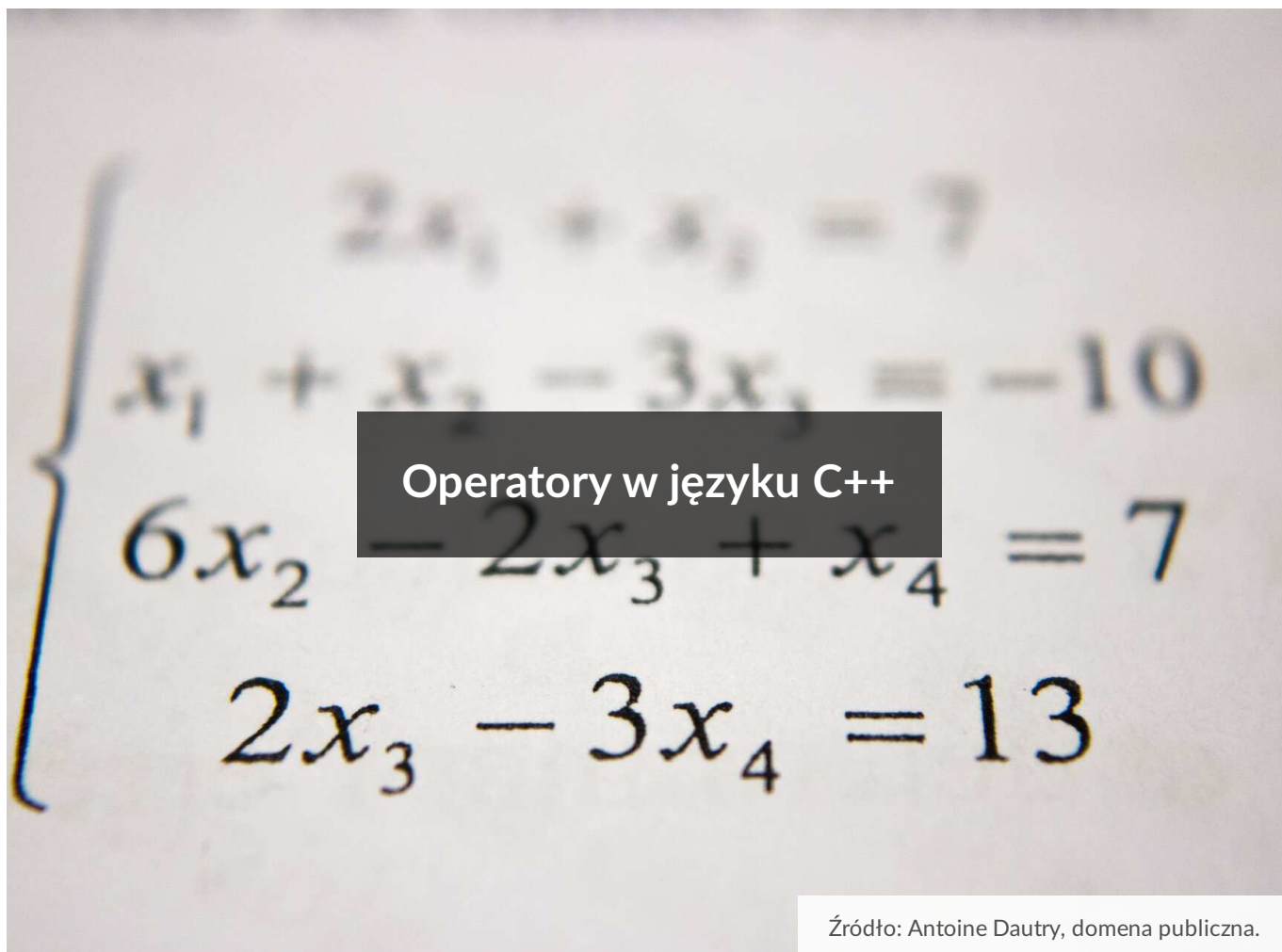


$$\begin{cases} 2x_1 + x_2 = 7 \\ x_1 + x_2 - 3x_3 = -10 \\ 6x_2 - 2x_3 + x_4 = 7 \\ 2x_3 - 3x_4 = 13 \end{cases}$$

Operatory w języku C++

- [Wprowadzenie](#)
- [Przeczytaj](#)
- [Audiobook](#)
- [Sprawdź się](#)
- [Dla nauczyciela](#)



Operatory to konstrukcje językowe zwracające pewne wartości. Poznaliśmy już kilka typów operatorów (np. arytmetyczne, logiczne, przypisań); wiemy również, czym się charakteryzują. Podstawowe cechy operatorów to m. in. liczba oraz typ argumentów, typ wyniku, wykonywane działanie, priorytet.

Pozostałe podstawowe informacje na ich temat znajdziesz w e-materiale [Operatory](#).

W tym e-materiale zapoznamy się z operatorami w języku C++.

Charakterystykę operatorów w innych językach programowania znajdziesz w e-materiałach:

- [Operatory w języku Java](#),
- [Operatory w języku Python](#).

Więcej zadań? Sięgnij do [Operatory – ćwiczenia](#).

Twoje cele

- Przeanalizujesz proces dobierania operatora w C++.
- Zbadasz wpływ operatorów na zmienne w C++.
- Napiszesz proste programy z wykorzystaniem poznanych operatorów.

Przeczytaj

Problem 1

Napisz prosty kalkulator. Do zmiennych a i b przypisz wartości całkowite, a następnie wykonaj kolejno działania:

- dodawania,
- odejmowania,
- mnożenia,
- dzielenia,
- dzielenia z resztą.

Specyfikacja:

Dane:

a, b - zmienne typu `integer`; $b \neq 0$.

Wynik:

Program wyświetla zaprogramowane działania.

Problem 2

Latami [przestępnymi](#) nazywamy te, których numeracja:

- jest podzielna przez 4 oraz jest niepodzielna przez 100;
- jest podzielna przez 400.

Napisz program sprawdzający, czy dany rok jest przestępny.

Specyfikacja:

Dane:

- rok – zmienna przechowująca badany rok; liczba naturalna

Wynik:

Na standardowym wyjściu podany w kodzie rok wyświetli się jako wartość 1 (przestępny) lub 0 (nieprzestępny).

Polecenie 1

Porównaj swoje rozwiązania z filmem.

Trwa wczytywanie danych ..



Wyrażenia arytmetyczne i logiczne

Operatory w C++



Film dostępny pod adresem </preview/resource/ReQClkCwrku1o>

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Film przedstawiający wyrażenia arytmetyczna oraz logiczne w języku C++.

Słownik

dekrementacja

zmniejszenie wartości o 1

inkrementacja

zwiększenie wartości o 1

instrukcja warunkowa

instrukcja wykonująca różne czynności, zależne od wyniku testu logicznego

rok przestępny

rok kalendarzowy, którego długość (365 dni) jest zwiększona w stosunku do lat nieprzestępnych (365 dni)

Audiobook

Polecenie 1

Zapoznaj się z audiobookiem. W parze z koleżanką lub kolegą przedyskutujcie, czy audiobook mógłby zostać uzupełniony innymi informacjami.

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

Operatory arytmetyczne

Ważne!

Deklarowanie zmiennej i przypisywanie jej wartości odbywa się następująco:

```
1 int a = 1;  
2 int b = 2;
```

Możemy również użyć znaku = w połączeniu z operatorami arytmetycznymi, aby tworzyć różne wariacje operatorów przypisania.

Jeżeli chcemy zmienić wartość zmiennej *a* przez dodanie do niej liczby 15, zrobimy to w następujący sposób:

```
1 a += 15;
```

W miejsce znaku + możemy wstawić inne operatory arytmetyczne. Ostatni fragment kodu odpowiada przypisaniu:

```
1 a = a + 15;
```

Ważne!

Zadanie inkrementacji i dekrementacji realizują podwójne operatory arytmetyczne:

```
1 a++; // gdy chcemy zwiększyć wartość zmiennej o 1
2 a--; // gdy chcemy zmniejszyć wartość zmiennej o 1
```

Operatory relacyjne oraz logiczne

Ważne!

Operator `||` zwraca wartość 1 (prawda), gdy przynajmniej jedno z wyrażeń jest prawdziwe; w przeciwnym przypadku zwracana jest wartość 0 (fałsz).

Porównajmy ze sobą dwa wyrażenia:

$$a > b \ || \ b > c.$$

Jeżeli a jest większe od b LUB b jest większe od c , to wynikiem operacji porównania będzie prawda. Fałsz otrzymamy tylko w przypadku, gdy oba wyrażenia są nieprawdziwe.

Operator `&&` zwróci prawdę wyłącznie w przypadku, gdy wszystkie wyrażenia są prawdziwe. W każdym innym przypadku zwrócony zostanie fałsz.

Zmieńmy operator logiczny z ostatniego przykładu na `&&`:

$$a > b \ \&\& \ b > c.$$

Jeżeli a jest większe od b ORAZ b jest większe od c , wyrażenie zwróci wartość PRAWDA. Wszystkie pozostałe kombinacje zwrócą FAŁSZ.

Dla zainteresowanych

W języku C++ wyrażenia logiczne sprawdzane są od lewej strony. Oznacza to, że program najpierw sprawdzi, czy a jest większe od b . Jeśli okaże się to prawdą, to

zwrócona zostanie prawda bez analizowania pozostałych warunków (w tym przypadku warunku $b > c$). Wynika to z tego, że dla operatora LUB wystarczy, aby co najmniej jeden z warunków był prawdziwy, aby zwrócił prawdę.

Jeśli natomiast okazałoby się, że a nie jest większe niż b , to warunki będą dalej sprawdzane albo do końca wyrażenia, albo odnalezienia warunku, który będzie prawdziwy.

Dla zainteresowanych

Analogicznie do operatora LUB operator ORAZ sprawdza warunki od lewej. Różnica polega na tym, że ORAZ zwraca fałsz przy pierwszym warunku, który okaże się nieprawdziwy.

Powodem jest to, że operator ORAZ zwraca fałsz, jeśli którykolwiek z warunków jest nieprawdziwy.

Ważne!

Zaprzeczenie zdania logicznego (czyli użycie operatora !) sprawia, że wyrażenie prawdziwe zwróci fałsz, a nieprawdziwe prawdę.

W wyniku zaprzeczenia:

$!(3 > 2)$

otrzymujemy fałsz, ponieważ trzy jest większe od dwóch.

Ważne!

Jeśli mamy dwie zmienne o nazwach a i b , których sumę chcemy zapisać we wcześniej zadeklarowanej zmiennej c , posłużymy się operatorem +:

1 $c = a + b;$

W przykładzie przypisaliśmy zmiennej c wartość sumy zmiennych a i b . Użyliśmy w tym celu operatora przypisania ($=$).

Ważne!

Do dyspozycji mamy sześć operatorów relacyjnych:

znak	znaczenie
$<$	mniejszy
$>$	większy
$<=$	mniejszy lub równy
$>=$	większy lub równy
$==$	równy
$!=$	różny

Gdy zdefiniujemy dwie zmienne a i b (a jest większe od b), zapiszemy następującą zależność:

$$a > b.$$

Możemy przedstawić również sytuację przeciwną:

$$a < b.$$

Operatorów relacyjnych używamy w testach logicznych (od których zależy działanie instrukcji warunkowych) i w połączeniu z operatorami logicznymi.

Polecenie 2

Przeanalizuj prezentację, a następnie, używając omówionych operatorów, spróbuj ułożyć własny test logiczny.

Polecenie 3

Sprawdź się

Pokaż ćwiczenia:   

Ćwiczenie 1



Ćwiczenie 2



Ćwiczenie 3



Ćwiczenie 4



Ćwiczenie 5



Ćwiczenie 6



Ćwiczenie 7



Ćwiczenie 8



Ćwiczenie 9



Dla nauczyciela

Autor: Zespół autorski Contentplus.pl sp. z o.o.

Przedmiot: Informatyka

Temat: Operatory w języku C++

Grupa docelowa:

Liceum ogólnokształcące i technikum, liceum ogólnokształcące, technikum, zakres podstawowy i rozszerzony

Podstawa programowa:

Cele kształcenia – wymagania ogólne

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Treści nauczania – wymagania szczegółowe

I. Rozumienie, analizowanie i rozwiązywanie problemów.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

6) objaśnia sposoby wykonywania przez komputer działań arytmetycznych i operacji logicznych;

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych.

Zakres podstawowy. Uczeń:

1) projektuje i programuje rozwiązania problemów z różnych dziedzin, stosuje przy tym: instrukcje wejścia/wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne, funkcje z parametrami i bez parametrów, testuje poprawność programów dla różnych danych; w szczególności programuje algorytmy z punktu I.2);

2) do realizacji rozwiązań problemów prawidłowo dobiera środowiska informatyczne, aplikacje oraz zasoby, wykorzystuje również elementy robotyki;

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

3) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu, uruchamianiu i testowaniu programów;

Kształtowane kompetencje kluczowe:

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

Cele operacyjne (językiem ucznia):

- Przeanalizujesz proces dobierania operatora w C++.
- Zbadasz wpływ operatorów na zmienne w C++.
- Napiszesz proste programy z wykorzystaniem poznanych operatorów.

Strategie nauczania:

- konstruktywizm;
- konektywizm.

Metody i techniki nauczania:

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimediu i ćwiczeń interaktywnych;
- ćwiczenia praktyczne.

Formy pracy:

- praca indywidualna;
- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

Środki dydaktyczne:

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda;
- oprogramowanie dla języka C++, w tym kompilator GCC/G++ 4.5 (lub nowszej wersji) i Code::Blocks 16.01 (lub nowszej wersji), Orwell Dev-C++ 5.11 (lub nowszej wersji) lub Microsoft Visual Studio.

Przebieg lekcji

Przed lekcją:

1. **Przygotowanie do zajęć.** Nauczyciel loguje się na platformie i udostępnia e-materiał: „Operatory w języku C++”. Uczniowie zapoznają się z treściami w sekcji „Przeczytaj”.

Faza wstępna:

1. Nauczyciel wyświetla uczniom temat, wskazuje cele zajęć oraz ustala z uczestnikami zajęć kryteria sukcesu.
2. **Rozpoznanie wiedzy uczniów.** Uczniowie tworzą pytania dotyczące tematu zajęć, na które odpowiedzą w trakcie lekcji.

Faza realizacyjna:

1. Nauczyciel w ramach sprawdzenia przygotowania uczniów do zajęć prosi wybrane osoby o przedstawienie rozwiązań problemu 1 i 2 z sekcji „Przeczytaj”. W razie potrzeby uczniowie ponownie oglądają film zamieszczony w tej sekcji.
2. **Praca z multimediami.** Nauczyciel przechodzi do sekcji „Audiobook”. Uczniowie w parach wykonują polecenie 1. Następnie dzielą się swoimi spostrzeżeniami na forum klasy. W kolejnym kroku indywidualnie zapoznają się z treściami dotyczącymi operatorów arytmetycznych, relacyjnych i logicznych.
3. **Ćwiczenie umiejętności.** Uczniowie wykonują indywidualnie ćwiczenia nr 1-8. Po wykonaniu każdego z nich następuje omówienie rozwiązania przez nauczyciela.

Faza podsumowująca:

1. Nauczyciel wyświetla na tablicy temat lekcji i cele zawarte w sekcji „Wprowadzenie”. W kontekście ich realizacji podsumowuje przebieg zajęć, a także wskazuje mocne i słabe strony pracy uczniów.
2. Wybrany uczeń podsumowuje zajęcia, zwracając uwagę na nabyte umiejętności.

Praca domowa:

1. Uczniowie wykonują polecenie 2 z sekcji „Audiobook”.
2. Uczniowie wykonują ćwiczenie 9 z sekcji „Sprawdź się”.

Materiały pomocnicze:

- Oficjalna dokumentacja techniczna dla języka C++.
- Oficjalna dokumentacja techniczna dla kompilatora GCC/G++ 4.5 (lub nowszej wersji).
- Oficjalna dokumentacja techniczna dla oprogramowania Code::Blocks 16.01 (lub nowszej wersji), Orwell Dev-C++ 5.11 (lub nowszej wersji) lub Microsoft Visual Studio.

Wskazówki metodyczne:

- Uczniowie mogą wykorzystać treści w sekcjach: „Przeczytaj”, „Audiobook”, „Sprawdź się” jako materiał do lekcji powtórkowej.