

Link do kompilatora C++ online https://www.onlinegdb.com/online_c++_compiler

DZIAŁANIA NA OPERATORACH MATEMATYCZNYCH

Operatorami matematycznymi nazywamy te znaki, które wykonują jakieś działanie matematyczne.

W C++ są to:

Dodawanie	+
Odejmowanie	-
Mnożenie	*
Dzielenie	/
Dzielenie modulo (reszta z dzielenia)	%

Zadanie 1. Dopisz do programu instrukcje wyjścia do konsoli (COUT) tekstu zawierającego ciąg obliczanych równań tak jak na przykładzie poniżej

```
9  #include <iostream>
10 using namespace std;
11 int main()
12 {
13     cout << "20 + 3 = " << 20 + 3 << endl;
14     cout << "20 - 3 = " << 20 - 3 << endl;
15     cout << "20 * 3 = " << 20 * 3 << endl;
16     cout << "20 / 3 = " << 20 / 3 << endl;
17     cout << "20 % 3 = " << 20 % 3 << endl;
18
19     return 0;
20 }
21
```

input

```
20 + 3 = 23
20 - 3 = 17
20 * 3 = 60
20 / 3 = 6
20 % 3 = 2
```

Wyjaśnienie wyników:

Dodawanie, odejmowanie i mnożenie zostało wykonane tak, jak się domyśliłeś. Natomiast wyjaśnić należy wynik działania dzielenia i dzielenia modulo.

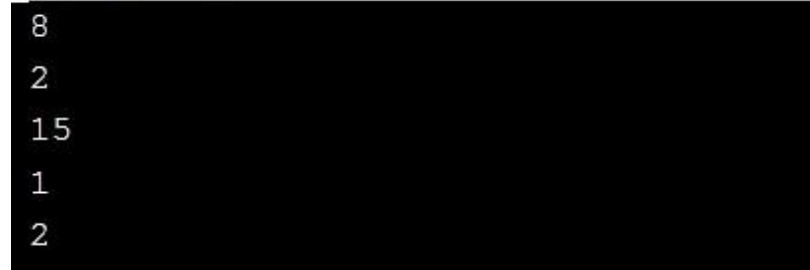
$20/3=6$ ponieważ nie został zadeklarowany typ zmiennej liczbowej, kompilator domyślnie przyjmuje wartości całkowite dlatego $20/3=6$ i reszty 2

Z kolei dzielenie modulo $20\%3=2$ ponieważ ten operator zwraca resztę z dzielenia czyli 2.

Zadanie 2. Działania matematyczne na zmiennych

Utwórz dwie zmienne X i Y przechowujące liczbę 5 i 3 tak jak na przykładzie poniżej. Uruchom program i sprawdź czy masz takie same wyniki.

```
9  #include <iostream>
10
11  using namespace std;
12
13  int main()
14  {
15      int x = 5;
16      int y = 3;
17
18      cout << x + y << endl;
19      cout << x - y << endl;
20      cout << x * y << endl;
21      cout << x / y << endl;
22      cout << x % y << endl;
23      return 0;
24  }
```



```
8
2
15
1
2
```

Kolejność wykonywania działań


Działania matematyczne w C++ wykonywane są od lewej do prawej. Jeśli chcesz zmienić kolejność wykonywania operacji, musisz wykorzystać nawiasy zaokrąglone. Oto przykład wykorzystujący powyższy fakt:

$$2+2*2 = 6$$

$$(2+2)*2 = 8$$

Wynik działań matematycznych zapisanych w kompilatorze

```
9  #include <iostream>
10
11  using namespace std;
12
13  int main()
14  {
15      cout << 2 + 2 * 2 << endl;
16      cout << ( 2 + 2 ) * 2 << endl;
17      return 0;
18  }
19
```



6
8

Zadanie 3. Dopisz instrukcje do programu z zadania 2, które obliczą równania matematyczne wypisane poniżej: (możesz je skopiować i wkleić do kompilatora)

$$x * x + x / y$$

$$x * (x + x) / y$$

$$2 * x - 1 * (y / y)$$

```
9  #include <iostream>
10 using namespace std;
11
12 int main()
13 {
14     int x = 5;
15     int y = 3;
16
17     cout << x + y << endl;
18     cout << x - y << endl;
19     cout << x * y << endl;
20     cout << x / y << endl;
21     cout << x % y << endl;
22     cout << x * x + x / y << endl;
23     cout << x * (x + x) / y << endl;
24     cout << 2 * x - 1 * (y / y) << endl;
25     return 0;
26 }
```

input

```
8
2
15
1
2
26
16
9
```

Zapis liczb zmiennoprzecinkowych w C++

Aby zapisać liczbę zmiennoprzecinkową np. 2,5 w języku C++ musisz użyć kropki zamiast przecinka. Wynika to z zastosowania amerykańskiego standardu zapisu liczb zmiennoprzecinkowych.

2.5 zapis poprawny

4,6 zapis niepoprawny

Zadanie 4. Dopisz 4 instrukcje do programu z zadania 3, które obliczą równania matematyczne wypisane poniżej oraz wyświetlą wyniki w konsoli:

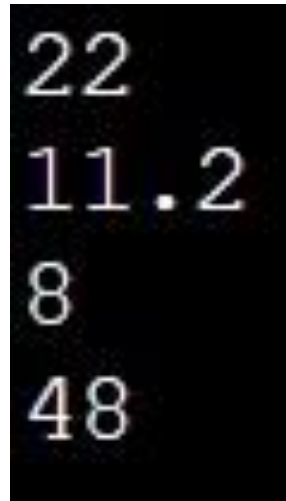
$$20 + x : 2$$

Wyniki

$$1,2 + 4 * y - 2$$

$$4 - x + 3^2$$

$$\frac{44 - 2}{2 + 4}$$



```
22
11.2
8
48
```

Obliczenia z wykorzystaniem zmiennych

Przykład obliczenia pola powierzchni i obwodu prostokąta na zmiennych.

Pamiętaj o poprawnej deklaracji typu zmiennej. Jeśli chcesz podawać liczby całkowite, użyj typu zmiennej INT.

Natomiast jeśli chcesz podawać liczby zmiennoprzecinkowe, użyć zmiennej typu FLOAT.

Zadanie 5. Przepisz program, skompiluj i uruchom. Sprawdź poprawność działania na różnych liczbach.

```
9  #include <iostream>
10
11 #include <iostream>
12
13 using namespace std;
14
15 int main()
16 {
17     int liczba1 = 3;
18     int liczba2 = 8;
19
20     cout<<"Pole powierzchni prostokąta wynosi "<< liczba1 * liczba2 <<endl;
21     cout<<"Obwód prostokąta wynosi "<< ( liczba1 * 2 ) + ( liczba2 * 2 ) <<endl;
22
23     return 0;
24 }
```

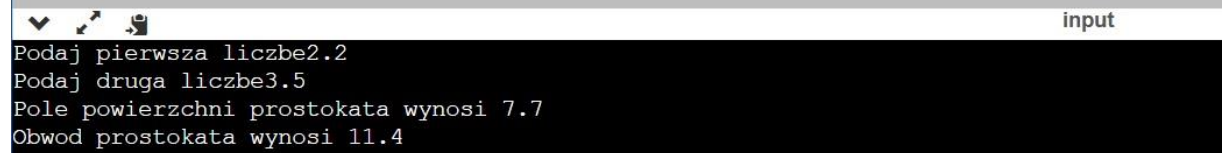
input

```
Pole powierzchni prostokąta wynosi 24
Obwód prostokąta wynosi 22
```

W powyższym programie można podawać tylko liczby całkowite typu INT, te liczby są wcześniej przypisane do zmiennych liczba1 i liczba2.

Zadanie 6. Zmień deklarację zmiennych z typu całkowitego na typ zmiennoprzecinkowy FLOAT. Następnie dopisz instrukcje, które będą prosić użytkownika o podanie w konsoli dwóch liczb. Przykład rozwiązania poniżej:

```
9  #include <iostream>
10
11  using namespace std;
12
13  int main ()
14  {
15      float liczba1 = 3;
16      float liczba2 = 8;
17      cout << "Podaj pierwsza liczbe";
18      cin >> liczba1;
19      cout << "Podaj druga liczbe";
20      cin >> liczba2;
21      cout << "Pole powierzchni prostokata wynosi " << liczba1 * liczba2 << endl;
22      cout << "Obwod prostokata wynosi " << (liczba1 * 2) + (liczba2 * 2) << endl;
23
24      return 0;
25  }
26
```



input

```
Podaj pierwsza liczbe2.2
Podaj druga liczbe3.5
Pole powierzchni prostokata wynosi 7.7
Obwod prostokata wynosi 11.4
```

Zadanie 7. Napisz program obliczający pole powierzchni trójkąta. Program ma prosić użytkownika o podanie długości podstawy trójkąta oraz jego wysokość. Następnie podać wynik pola powierzchni trójkąta.

Skorzystaj z kodu programu napisanego w zadaniu 6.

Zadanie 8. Pole powierzchni koła i obwód

Zadanie 9. Pole powierzchni kuli


Kolejność wykonywania instrukcji w C++

Program napisany w C++ jest wykonywany instrukcja po instrukcji, tak jak czytany tekst w książce (czyli od lewej do prawej, linijka po linijce od góry do dołu).

Instrukcją nazywamy polecenie programie, które zakończone jest średnikiem ;
np.

```
int wiek = 16;  
cout << imie << endl;
```

```
1  #include <iostream> ← 1. Dołączanie bibliotek  
2  
3  using namespace std; ←  
4  
5  int main() ← 2. Uruchomienie funkcji głównej main ()  
6  {  
7      string imie = "Adam"; ← 3. Deklaracja zmiennej imie i przypisanie jej wartości.  
8      int wiek = 16; ← 4. Deklaracja zmiennej wiek i przypisanie jej wartości  
9  
10     cout << "Witaj " << imie << endl; ← 5. Pobranie zmiennej imie i wypisanie jej w konsoli  
11     cout << "Masz " << wiek << " lat"; ← 6. Pobranie zmiennej wiek i wypisanie jej w konsoli  
12  
13     return 0; ← 7. Program wykonał poprawnie wszystkie instrukcje i zwraca  
14 }      wartość całkowitą 0 do funkcji głównej main ().  
15
```



Wynik kodu.

```
Witaj Adam  
Masz 16 lat
```