Выполнил студент группы 821703 Морозов А.Ю.

Pomidor Java Language 2021 (PJL) - язык для работы с численными данными.

Типы данных

int- целочисленный тип данных. Поддерживаемые значения : -2 147 483 648 до 2 147 483 647. Затраты памяти : 4 байта.
float, double - численный тип данных, поддерживающий как целый числа, так и числа с плавающей точкой .

Типы с плавающей точкой

Тип	Размер (бит)	Диапазон
float	32	от 1.4e-45f до 3.4e+38f
double	64	от 4.9e-324 до 1.7e+308

Операции

Сложение - “+” операция сложения чисел. Является бинарной операцией.
Пример:
int a = 20
int b = 10
double c = 1488
a = b + c
Результат:
a = 1498
Возвращаемый тип эквивалентен типу первого слагаемого.
Операция поддерживает любые доступные типы.

Присвоение и сложение - “+=” операция сложения, при которой результат присваивается первому слагаемому. Является унарной операцией.
Пример:
int a = 0
int b = 10

a += b
Результат:
a = 10
Возвращаемый тип эквивалентен типу первого слагаемого.
Операция поддерживает любые доступные типы.

Вычитание - “-” операция вычитания чисел. Является бинарной операцией.
Пример:
float a = 2.2
float b = 10.1
float c = 0
c = b - a
Результат:
c = 7.9
Возвращаемый тип эквивалентен типу уменьшаемого.
Операция поддерживает любые доступные типы.

Присвоение и вычитание- “-=” операция вычитания, при которой результат присваивается уменьшаемому. Является унарной операцией.
Пример:
float a = 2.2
float b = 10.1
b -= a
Результат:
b = 7.9
Возвращаемый тип эквивалентен типу уменьшаемого.
Операция поддерживает любые доступные типы.

Умножение- “*” операция умножения чисел. Является бинарной операцией.
Пример:
int a = 10
int b = 100
double c = 0
c = b * a
Результат:
a = 110
Возвращаемый тип эквивалентен типу множимого.
Операция поддерживает любые доступные типы.

Присвоение и умножение - “*=” операция умножения, при которой результат присваивается множимому. Является унарной операцией.
Пример:
int a = 10
int b = 100
b *= a
Результат:
b = 1000
Возвращаемый тип эквивалентен типу множимого.
Операция поддерживает любые доступные типы.

Деление- “/” операция деления чисел. Является бинарной операцией.
Пример:
int a = 100
int b = 30
int c = 0
c = b / a
Результат:
c = 3
Возвращаемый тип эквивалентен типу делимого.
Операция поддерживает любые доступные типы.

Присвоение и деление- “/=” операция вычитания, при которой результат присваивается делимому. Является унарной операцией.
Пример:
int a = 100
int b = 30
a /= b
Результат:
a = 3
Возвращаемый тип эквивалентен типу делимого.
Операция поддерживает любые доступные типы.

Возведение в степень- “^” операция возведения числа в степень. Является бинарной операцией.
Пример:
int a = 3
int b = 2
int c = 0
c = a ^ b

Результат:
c = 9
Возвращаемый тип эквивалентен типу числа, которое возводят в степень.
Операция поддерживает **int** и **double**.

Присвоение - "=" операция присвоения значения переменной. Является унарной операцией.
Пример:
int a = 0
int b = 9
a = (b + 3) *2
Результат:
b = 24
Возвращаемый тип эквивалентен типу присваиваемого числа.
Операция поддерживает все доступные типы.

Выражения(Операторы)

for - цикл, который может содержать в своем теле любые операции или выражения.
В скобках цикла указывается:

1. Название переменной, которая используется в условиях цикла
2. Условие, если которое верное, то тело цикла выполняется
3. Изменение, число, на которое изменяется переменная цикла

.
Пример:
```
int b = 1
for(int a=0; a < 10; 1):
        b += 1
```

Результат:
Пройдёт десять итераций цикла.
b = 11

while - цикл, который содержит условие, если которое верное, то тело цикла выполняется
.
Пример:
```
int b = 1
while(b<100):
```

b += 2

Результат:
Пройдёт 100 итераций цикла.
b = 101

if - выражение условия, который может содержать в своём теле любые операции или выражения.
В скобках должно указываться условие, при котором будет выполняться тело выражения.
В условии могут использоваться как переменные, так и значения.
Пример:
int a = 0
int b = 1
if(a < 10):
b += 10

Результат:
Условие верное, тело выполнится
b = 11

Ошибки

Error 1 : Illegal variable used - ошибка обозначающая, что используемая переменная не объявлена.

Пример:
int a = 0
int b= 1
a += c
Результат:
Illegal variable used: c

Error 2 : used variable already defined - ошибка означающая, что вы хотите объявить переменную, которая уже объявлена.

Пример:
int a = 0
int a = 1

Результат:
Used variable already defined: a

Error 4 : Illegal number of function parameters - ошибка означающая, что в функцию передано неправильное количество параметров.

Пример:
INT FUNC (int name1, int name2):
//
…
//

{
int a = 0
FUNC(a)
}
Результат:
Illegal number of function parameters

Error 5 : Syntax error - синтаксическая ошибка в коде

Пример:
{
int a = 0
&&&

}
Результат:
Syntax error
Rule stack: line: 3

Операторы сравнения

< - Оператор меньше.

Пример:
int a = 0
int b = 1
a < b
Результат:
true

<= - Оператор меньше или равно.

Пример:
int a = 10
int b = 10
a <= b
Результат:
true

> - Оператор больше.

Пример:
float a = 2.2
float b = 2.1
a > b
Результат:
true

>= - Оператор больше или равно.

Пример:
float a = 2.05
float b = 2.1
a >= b
Результат:
false

== - Оператор равенства.
Пример:
int a = 1488
int b = 1488
a == b
Результат:
true

!= - Оператор неравенства.

Пример:
int a = 1488

int b = 1488
a != b
Результат:
false

Для всех операторов сравнения поддерживаются все доступные типы.

Преобразование типов

Поддерживается неявное преобразование типов.
Пример:
int a = 2
float b = 2.2
a = b
Результат:
a = 2

Оператор break

Оператор **break** завершает выполнение функции и возвращает управление вызывающей функции. Выполнение возобновляется в вызывающей функции в точке сразу после вызова

Пример:
INT FUNC (int name1, int name2):
break name1 + name2

{
int a = 5
int b = FUNC(a, 10)
}

Оператор Print

Оператор **print** используется для вывода данных на экран (в консольное окно).

Пример:
{
int a = 5
int b = 100
print((a+b)^2)
}

Результат:
Выведет в консоль 11 025