**Мелешко Алексей Сергеевич, 921704**

Язык, описывающий математические вычисления

**Типы данных:**

**int -** целочисленный тип данных.

Поддерживаемые значения : -2 147 483 648 до 2 147 483 647.

Затраты памяти : 2 байта. **float-** численный тип данных, поддерживающий как целый числа, так и числа с плавающей точкой .

Поддерживаемые значения:

1.Целые числа: -3.4 \* 10^38 до 3.4 \* 10^38 2.Числа с плавающей точкой: до 3.4 \* 10^-38 Затраты памяти : 4 байта.

**Объявление переменных:**

**Объявление типа int. Пример:**

int a = 1; int b = 5;

**Объявление типа float.**

**Пример:**

float a = 1.01; float b = 5.9;

**Операции:**

**Сложение - “+”** операция сложения чисел. Является бинарной операцией. **Пример**:

int a = 0; int b = 10; int c = 1488; a = b + c; **Результат:** a = 1498;

Операция поддерживает числовые(int float) типы.

**Вычитание - “-”** операция вычитания чисел. Является бинарной операцией. **Пример**:

float a = 2.2; float b = 10.1; float c = 0.0; c = b - a;

**Результат:**

c = 7.9;

Операция поддерживает числовые(int float) типы.

**Умножение- “\*”** операция умножения чисел. Является бинарной операцией. **Пример**:

int a = 10; int b = 100; int c = 0; c = b \* a;

**Результат:** a = 110;

Операция поддерживает числовые(int float) типы.

**Деление- “/”** операция деления чисел. Является бинарной операцией. **Пример**:

int a = 100; int b = 30; int c = 0; c = b / a;

**Результат:**

c = 3;

Операция поддерживает числовые(int float) типы.

**Присвоение - “=”** операция присвоения значения переменной. Является унарной операцией. **Пример**:

int a = 0; int b = 9; a = b;

**Результат:** a = 9;

Операция поддерживает все доступные типы.

**Выражения: for -** цикл, который может содержать в своем теле любые операции или выражения. **Пример:**

int a = 1; for(int i = 0;i<10;i=i+1){ a = a + 1;

}

**Результат:** a = 10;

**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 = 100; if(a<2000){ a = 2000;

}

**Результат**: a = 2000;

**Ошибки:**

**Error 1 : variable is duplicated -** ошибка означающая, что вы хотите объявить переменную, которая уже объявлена.

**Пример**:

float a; int a = 0;

**Error 2:provided type: -** ошибка означающая, что в вернуть из метода меременную или литерал типа, не соответствующего типу возврата метода. **Пример**: int sum(){ return 1.0;

}

**Error 3:expected type: -** ошибка означающая, что пытаетесь инициализировать переменную литералом не того типа. **Пример**:

int q = “sds”;

**Операторы сравнения: < -** Оператор меньше.

**Пример**:

a = 0; b = 1; a < b;

**Результат: true**

**<= -** Оператор меньше или равно.

**Пример**:

a = 10; b = 10; a <= b; **Результат: true**

**> -** Оператор больше.

**Пример**: a = 2.2; b = 2.1; a > b;

**Результат: true**

**>= -** Оператор больше или равно.

**Пример**: a = 2.05; b = 2.1; a >= b; **Результат: false**

**== -** Оператор равенства. **Пример**:

a = 1488; b = 1488;

a == b; **Результат: true**

**!= -** Оператор неравенства.

**Пример**: a = 1488; b = 1488; a != b;

**Результат: false**

Для всех операторов сравнения поддерживаются все типы:int, float.