**Оглавление**

[**ФОРМУЛА** 2](#_Toc179455883)

[**ТАБЛИЦА** 3](#_Toc179455884)

[**ТЕКСТ** 4](#_Toc179455885)

[**РИСУНОК** 5](#_Toc179455886)

[**СПИСОК ЛИТЕРАТУРЫ** 6](#_Toc179455887)

**ФОРМУЛА**

**ТАБЛИЦА**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Столбец 1 | Столбец 2 | Столбец 3 | Столбец 4 | Столбец 5 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**ТЕКСТ**

C# является строго типизированным языком. Каждая переменная и константа имеет тип, как и каждое выражение, результатом вычисления которого является значение. Каждое объявление метода задает имя, тип и вид (значение, ссылка или вывод) для каждого входного параметра и для возвращаемого значения.

В библиотеке классов .NET определены встроенные числовые типы и комплексные типы, представляющие разнообразные конструкции. К ним относятся файловая система, сетевые подключения, коллекции и массивы объектов, а также даты. Обычная программа на C# использует типы из этой библиотеки классов и пользовательские типы, которые моделируют уникальные концепции конкретной сферы применения.

В типах может храниться следующая информация:

* место, необходимое для хранения переменной этого типа;
* максимальное и минимальное значения, которые могут быть представлены;
* содержащиеся в типе члены (методы, поля, события и т. д.);
* базовый тип, от которого наследует этот тип;
* реализуемые им интерфейсы;
* разрешенные виды операций.

Компилятор использует сведения о типах, чтобы проверить, все ли операции, выполняемые в коде, являются типобезопасными. Например, при объявлении переменной типа [int](https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/builtin-types/integral-numeric-types) компилятор позволяет в дополнение использовать переменную и операции вычитания. При попытке выполнить эти же операции для переменной типа [bool](https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/builtin-types/bool) компилятор выдаст ошибку.

Компилятор внедряет сведения о типе в исполняемый файл в виде метаданных. Среда CLR использует эти метаданные во время выполнения для дальнейшего обеспечения безопасности типа при выделении и освобождении памяти.

**РИСУНОК**

Начало

Ввод чисел a и b

Нахождение суммы

Конец

Рисунок 1 – Подпись прямо под рисунок. Прямо по центру под рисунком

**СПИСОК ЛИТЕРАТУРЫ**

1. Герберт Ш. C++. Базовый курс.
2. Фленов М. Е. Библия C#. 6-е изд.