<article id="part1"> История создания</article>

<p>  
Язык программирования С# был разработан в 1993—2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft.

<p>К 2000 году у Microsoft были готовы промышленные версии новых технологий и решений для обмена сообщениями и данными, а также для создания Internet-приложений. Была выпущена и новая платформа для разработки под новые решения — .NET. В ней объединились сразу несколько языков программирования, что было в новинку для того времени.

<p>Ещё одним новшеством платформы .NET была технология активных серверных страниц ASP.NET (Active Server Page). С её помощью можно было относительно быстро разработать веб-приложения, взаимодействующие с базами данных. Специально для ASP.NET был создан язык программирования C#. Да и сама ASP.NET была полностью написана на нём. <p><p>

<article id="part2"> Краткий обзор языка</article>

<p>C# — это язык с C-подобным синтаксисом. Здесь он близок в этом отношении к C++ и Java.

<p>Будучи объектно-ориентированным языком, он много перенял у Java и С++. Как и Java, C# изначально предназначался для веб-разработки, и примерно 75% его синтаксических возможностей такие же, как у  Java. C# также называют «очищенной версией Java». Ещё 10% наш герой позаимствовал из C++ и 5% – из Visual Basic. Оставшиеся 10% C# — это реализация собственных идей разработчиков. Объектно-ориентированный подход позволяет строить с помощью C# крупные, но в то же время гибкие, масштабируемые и расширяемые приложения.

<p>C# уже давно поддерживает много полезных функций:<ul><li>инкапсуляция, </li><li>наследование,<li>полиморфизм,</li> <li>перегрузка операторов,</li><li>статическая типизация.</li></ul>

<p>У «шарпа» выделяют много преимуществ:<p><ul class=icon\_img><li>Поддержка подавляющего большинства продуктов Microsoft. </li><li>Бесплатность ряда инструментов для небольших компаний и некоторых индивидуальных разработчиков — Visual Studio, облако Azure, Windows Server, Parallels Desktop для Mac Pro и др. </li><li>Типы данных имеют фиксированный размер (32-битный int и 64-битный long), что повышает «мобильность» языка и упрощает программирование, так как вы всегда знаете точно, с чем вы имеете дело. </li><li>Автоматическая «сборка мусора»  Это значит, что нам в большинстве случаев не придётся заботиться об освобождении памяти. Вышеупомянутая общеязыковая среда CLR сама вызовет сборщик мусора и очистит память. </li><li>Большое количество «синтаксического «сахара» — специальных конструкций, разработанных для понимания и написания кода. Они не имеют значения при компиляции. </li><li>Низкий порог вхождения. Синтаксис C# имеет много схожего с другими языками программирования, благодаря чему облегчается переход для программистов. Язык C# часто признают наиболее понятным и подходящим для новичков. </li><li>С помощью Xamarin на C# можно писать программы и приложения для таких операционных систем, как iOS, Android, MacOS и Linux. </li><li>Сегодня в любом регионе России имеется немало вакантных мест на должность C#-программиста.</li></ul><p>Но есть у C# и некоторые недостатки:<p><ul class=”icon\_img”><li>Приоритетная ориентированность на платформу Windows; </li><li>Язык бесплатен только для небольших фирм, индивидуальных программистов, стартапов и учащихся . Крупной компании покупка лицензионной версии этого языка обойдётся в круглую сумму. </li></ul>

<p><article id="part3"> Краткий обзор языка</article>

<p>

<span class="IDE-clr-blue"> using </span> System;

<br>

<span class="IDE-clr-blue">namespace </span> Code\_Example<br>

{<br>

<span class="IDE-clr-blue">internal class </span> <span class="IDE-clr-lblue">Program </span>

<br> {<br>

<br> <span class="IDE-clr-blue">static void </span> Main(<span class="IDE-clr-blue">string </span>[] args) <br>

<br>{<br>

Console.WriteLine("Программа для вычисления площади, высоты, радиусов вписанной и описанной окружности\nравностороннего треугольника. Введите значение строны треугольника:");

<br> <span class="IDE-clr-blue">double </span> a = Convert.ToDouble(Console.ReadLine());

<br> <span class="IDE-clr-blue">if </span> (a != 0)

<br> {<br>

<br> Console.WriteLine("Какой параметр необходимо вычислить? Введите 'площадь', 'высоту',\n'радиус вписанной окружности' или 'радиус описанной окружности'.");

<br> <span class="IDE-clr-blue">string </span> choice = Console.ReadLine();

<br> <span class="IDE-clr-blue">switch </span> (choice)

<br>{

<br> <span class="IDE-clr-blue">case </span> "площадь":

<br> <span class="IDE-clr-blue">double </span> s = Math.Sqrt(3) / 4 \* Math.Pow(a, 2);

<br>Console.WriteLine($"Площадь равна: {s}");

<br> <span class="IDE-clr-blue">break </span>;

<br> <span class="IDE-clr-blue">case </span> "высоту":

<br> <span class="IDE-clr-blue">double </span> h = Math.Sqrt(3) / 2 \* a;

<br>Console.WriteLine($"Высота равна: {h}");

<br> <span class="IDE-clr-blue">break </span>;

<br> <span class="IDE-clr-blue">case </span> "радиус вписанной окружности":

<br> <span class="IDE-clr-blue">double </span> r = a / (2 \* Math.Sqrt(3));

<br>Console.WriteLine($"Радиус вписанной окружности: {r}");

<br> <span class="IDE-clr-blue">break </span>;

<br> <span class="IDE-clr-blue">case "радиус описанной окружности":

<br> <span class="IDE-clr-blue">double </span> R = 2 / Math.Sqrt(3) \* a;

<br> Console.WriteLine($"Радиус описанной окружности: {R}");

<br> <span class="IDE-clr-blue">break </span>;

<br> <span class="IDE-clr-blue">default </span>:

<br>Console.WriteLine("Ошибка ввода, возможно вы ввели неверное значение.\n\nПерезапустите программу и попробуйте снова.");

<br> <span class="IDE-clr-blue">break </span>;

<br>}

<br>}

<br> <span class="IDE-clr-blue">else </span>

<br>{

<br>Console.WriteLine("Некорректное значение длины, перезапустите программу");

<br>}

<br> Console.ReadKey();

<br>}

<br> }

<br>}