Language Map for C# by Steven Blaine

Variable Declaration

Is this language strongly typed or dynamically typed? Provide an example of how variables are declared in this language.

C# is a strongly-typed language.

Two examples of variable declarations:

```
int faveYear = 1987;
string university = “Bellarmine”;
```

Data Types

List all of the data types (and ranges) supported by this language.

DATA TYPE	RANGE
sbyte	-128 to 127
byte	0 to 255
short	-32,768 to 32,767
ushort	0 to 65,535
int	-2,147,483,648 to 2,147,483,647
uint	0 to 4,294,967,295
long	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
ulong	0 to 18,446,744,073,709,551,615
nint	(DEPENDS ON PLATFORM)
nuint	(DEPENDS ON PLATFORM)
float	$-\pm 1.5 \times 10^{-45}$ to $\pm 3.4 \times 10^{38}$
double	$\pm 5.0 \times 10^{-324}$ to $\pm 1.7 \times 10^{308}$

	decimal	±1.0 x 10^{-28} to ±7.9228 x 10^{28}
	char	ANY VALID CHARACTER (*e.g.*, a,*, \x0058 (hex), or\u0058 (Unicode))
	bool	TRUE or FALSE
	string	SEQUENCE OF UNICODE CHARACTERS
	object	(BASE TYPE OF OTHER TYPES)
	DateTime	0:00:00am 1/1/01 to 11:59:59pm 12/31/9999

Selection Structures

Provide examples of all selection structures supported by this language (if, if else, etc.)

C# selection structures are “if,” if else,” and “switch.”

```
// IF EXAMPLE
DisplayMeasurement(45);  // Output: The measurement value is 45
DisplayMeasurement(-3);  // Output: Warning: not acceptable value! The measurement value is -3

void DisplayMeasurement(double value)
{
   if (value < 0 || value > 100)
   {
      Console.Write("Warning: not acceptable value! ");
   }

   Console.WriteLine($"The measurement value is {value}");
}

***************

// IF ELSE EXAMPLE

DisplayWeatherReport(15.0);  // Output: Cold.
DisplayWeatherReport(24.0);  // Output: Perfect!

void DisplayWeatherReport(double tempInCelsius)
{
   if (tempInCelsius < 20.0)
```

```
    {
        Console.WriteLine("Cold.");
    }
    else
    {
        Console.WriteLine("Perfect!");
    }
}


***************

// SWITCH EXAMPLE

DisplayMeasurement(-4);  // Output: Measured value is -4; too low.
DisplayMeasurement(5);  // Output: Measured value is 5.
DisplayMeasurement(30);  // Output: Measured value is 30; too high.
DisplayMeasurement(double.NaN);  // Output: Failed measurement.

void DisplayMeasurement(double measurement)
{
    switch (measurement)
    {
        case < 0.0:
            Console.WriteLine($"Measured value is {measurement}; too low.");
            break;

        case > 15.0:
            Console.WriteLine($"Measured value is {measurement}; too high.");
            break;

        case double.NaN:
            Console.WriteLine("Failed measurement.");
            break;

        default:
            Console.WriteLine($"Measured value is {measurement}.");
            break;
    }
}
```

Repetition Structures
Provide examples of all repetition structures supported by this language (loops, etc.)
	C# repetition structures are “for,” “foreach,” “while,” and “do while.”
	
	// FOR EXAMPLE
	
	for (int i = 0; i < 3; i++)
	{
	  Console.Write(i);
	}
	// Output:
	// 012
	
	**********
	
	// FOREACH EXAMPLE
	
	var fibNumbers = new List<int> { 0, 1, 1, 2, 3, 5, 8, 13 };
	foreach (int element in fibNumbers)
	{
	  Console.Write($"{element} ");
	}
	// Output:
	// 0 1 1 2 3 5 8 13
	
	**********
	
	// WHILE EXAMPLE
	
	int n = 0;
	while (n < 5)
	{
	  Console.Write(n);
	  n++;
	}
	// Output:
	// 01234
	
	**********
	
	// DO WHILE EXAMPLE

<table>
<tr><td></td><td>

```
int n = 0;
do
{
   Console.Write(n);
   n++;
} while (n < 5);
// Output:
// 01234
```

</td></tr>
<tr><td>Arrays
If this language supports arrays, provide an example of creating an array with a primitive data type (e.g. float, int, etc.)</td><td>C# does support arrays. An example:

// Declare a single-dimensional array of 5 integers
int[] array1 = new int[5] { 10, 2, 4, 12, 1 };</td></tr>
<tr><td>Data Structures
If this language provides a standard set of data structures, provide a list of the data structures and their Big-Oh complexity.</td><td>
<table>
<tr><th>DATA STRUCTURE</th><th>BIG-O TIME (NOT SPACE) COMPLEXITY (WORST CASE – NOT NECESSARILY AVERAGE TIME COMPLEXITY)</th></tr>
<tr><td>Array

[Note ArrayList is generally same as Array except its size increases dynamically; List is a generic implementation of ArrayList]</td><td>Access: O(1)
Search: O(n)
Insertion: O(n) (append: O(1))
Deletion: O(n)</td></tr>
<tr><td>Linked list (singly)</td><td>Access: O(n)
Search: O(n)
Insertion: O(1)
Deletion: O(1)</td></tr>
<tr><td>Queue</td><td>Access: O(n)
Search: O(n)
Insertion: O(1)
Deletion: O(1)</td></tr>
<tr><td>Stack</td><td>Access: O(n)
Search: O(n)
Insertion: O(1)
Deletion: O(1)</td></tr>
</table>
</td></tr>
</table>

	Hash table	Access: O(n) Search: O(n) Insertion: O(n) Deletion: O(n)
	Graph	Varies depending on the choice of algorithm -- O(n*lg(n)) or slower for most graph algorithms
	Tree	Worst time complexity varies between O(log(n)) and O(n)
	Binary search tree	Access: O(n) Search: O(n) Insertion: O(n) Deletion: O(n)
	PriorityQueue / Heap	For retrieval methods, it has constant time complexity Insertion: O(log(n)) Deletion: O(log(n))
	Binary search	Can use sequential search with O(n) complexity, or binary search with O(log n) complexity if the elements are sorted
	Dictionary	Generally, Dictionary is a generic implementation of Hash table; searching in a Dictionary has the complexity of time O(1)
	SortedList	Lookup efficiency for Key is O(log n)

Objects

If this language support object-orientation, provide an example of how to create a simple object with a default constructor.

C# is an object-oriented programming language.

A constructor with no parameters is called a default constructor.

Object example:

```
public class Taxi
{
   public bool IsInitialized;

   public Taxi()
   {
      IsInitialized = true;
   }
```

	}
	
	class TestTaxi
	{
	  static void Main()
	  {
	  Taxi t = new Taxi();
	  Console.WriteLine(t.IsInitialized);
	  }
	}

Runtime Environment *What runtime environment does this language compile to? For example, Java compiles to the Java Virtual Machine.* *Do other languages also compile to this runtime?*	The .NET runtime. The .NET framework can work with a number of other programming languages, with prominent ones including B.NET, C++, and F#.
Libraries/Frameworks *What are the popular libraries or frameworks used by programmers for this language? List at least three (3).*	Three sample libraries (found on GitHub): • Newtonsoft / Json.NET • Cake • Xunit
Domains *What industries or domains use this programming language? Provide specific examples of companies that use this language and what they use it or.*	C# is widely used for developing desktop applications, web applications, and web services. Some companies using C# in their tech stacks include: • Accenture (professional services and tech consulting) • General Motors (automotive industry) • Uber (mobile device-based ridesharing) • Unity (gaming software)