

Centro Universitário 7 de Setembro
Programação para Plataformas Proprietárias I
Prof. Fernando Vasconcelos Mendes
Exercício Teórico

Aluno: **Raul Figueira Miranda**

1. O que é a Common Language Runtime (CLR) ? Por que ela é uma parte importante do .NET Framework?

É um ambiente que gerencia a execução do código e que oferece alguns serviços. É importante pois oferece, por exemplo, diversos recursos como integração e tratamento de exceção entre linguagens, coletor de lixo e compilador de linguagem intermediária para código nativo (compilador Just-In-Time). Ao fazer um paralelo ao Java, pode-se dizer que a CLR funciona como uma máquina virtual, mas voltada para a plataforma .NET.

2. Descreva o processo de compilação de um programa C# para um assembly e o processo de execução de um assembly.

O código (em C#, por exemplo), após ser compilado, gera um assembly gerenciado (com extensão .exe ou .dll) que é constituído de uma linguagem intermediária (MSIL) carregada pela Common Language Runtime. A CLR, então, trata de converter esse assembly em um código de máquina nativo por meio de um compilador Just-In-Time (JIT). Na execução, a CLR carrega o código intermediário, verifica se o mesmo está correto e o executa. Durante o processo de execução do código gerenciado, é recebido serviços como o coletor de lixo, controle de versão e segurança. Assim, esse processo fica sendo constantemente monitorado para garantir a qualidade em gerenciamento otimizado de memória, segurança e performance.

3. O que é a linguagem MSIL ? Aponte algumas características chaves.

MSIL é a abreviação para Microsoft Intermediate Language e é o resultado da compilação de um código fonte de uma linguagem .NET de mais alto nível (como C#). A MSIL, ou simplesmente IL, é um conjunto de instruções de baixo nível que são independentes de CPU, que pode ser entendida pela CLR e que pode ser convertida em código nativo por um compilador JIT. Ao fazermos um paralelo com Java, pode-se dizer que a MSIL é como se fosse o bytecode que é rodado pela máquina virtual do Java.

4. O que é a Common Language Specification (CLS) ? Por que ela foi desenvolvida?

O .NET Framework independente de linguagem, assim, é possível utilizá-lo programando em C#, Eiffel, Visual Basic, entre outras. Para que isso seja possível, é necessário que essas linguagens sigam determinadas especificações. A CLS é, portanto, um conjunto de regras que se aplicam aos assemblies gerados de tal maneira que o .NET Framework

se torne independente da linguagem que esteja sendo usada, promovendo, inclusive a interação entre módulos escritos em linguagens diferentes, pois, no final das contas, todas acabam se transformando em uma única linguagem intermediária, tornando possível a comunicação de códigos inicialmente diversos. Quando uma linguagem adere às especificações, ela pode fazer parte da plataforma, assim, passa a ser nomeada com o .NET ao final. Por exemplo, Python ao aderir a CLS passa a ser nomeada de Python.NET.

5. O que é o Common Type System? Quando ele é utilizado?

É a definição de tipos de dados do .NET Framework. Tudo é um objeto e deriva da classe System.Object. Existe uma organização do sistema de tipos de dados de duas maneiras: tipo valor e tipo referência. Variáveis do primeiro tipo são alocadas na stack e derivam de System.ValueType, enquanto que as variáveis de referência são alocadas na heap e tem como classe base a System.Object. Assim, o CTS define como os tipos são declarados, usados e gerenciados no CLR e, além disso, serve de suporte para integração entre linguagens.