**QUESTÕES TEÓRICAS 1,5,7,9**

**-Questão 1**

A linguagem de alto nível é a que a sintaxe (forma de escrita) está mais próxima da linguagem humana do que a da máquina. Já a de baixo nível é a que a sintaxe mais se aproxima a linguagem da máquina.

A principal diferença dentre as duas, é que a de alto nível de mais fácil compreensão, pois é abstraído da sintaxe o que você não precisa entender em primeiro contato, como as instruções para o computador processar um simples “hello world”. As de mais baixo nível, você precisa ter o conhecimento da arquitetura do computador para utiliza-las.   
 Porém essas diferenças não querem dizer que umas são melhores que as outras, vai depender da necessidade do programador.

\*Exemplo de linguagens de alto nível: JavaScript, Python, PHP, Ruby.  
\*Exemplo de linguagem de baixo nível: Assembly.

O compilador é importante no processo da criação de um software, pois é ele quem vai “traduzir” a linguagem textual, escrita pelo programador, para a linguagem da máquina. Porém também é comum compiladores que geram o código pra uma máquina virtual que vai ser interpretado por um interpretador.

**-Questão 5**

* Linguagens imperativas: A programação imperativa é definida por ações e sequencias de comandos que manipulam o valor das variáveis, através de atribuições utilizando células de memória.

Exemplos de linguagem imperativa: **PASCAL, C**.

* Linguagens funcionais: A programação funcional é toda baseada em funções, onde o programador define a função para o computador para ser resolvido um problema, e o computador age como uma calculadora que avalia as expressões, simplificando até chegar ao resultado

Exemplos de linguagem funcional: **LISP, Harskell**.

* Linguagens lógicas: A programação lógica faz uso da lógica matemática, a qual é baseada na lógica simbólica. Usa o processo de inferência lógica para produzir os resultados. Possui a sintaxe bem diferente das linguagens funcionais e imperativas.

Exemplos de linguagem lógica: **Planner, Prolog**.

* Linguagens Marcação/Híbrida: São linguagens que com um conjunto de códigos aplicados a um texto ou a dados, para distingui-lo sintaticamente, adicionando particularidades sobre eles.

Exemplos de linguagem de Marcação/Híbrida: **HTML, XML**.

**-Questão 7**

Um paradigma de programação, é um conjunto de características que dão uma identidade particular a uma linguagem, que define a forma como ela pode abordar e resolver determinado problema, através de soluções cabíveis as suas características. Existem linguagens de programação com mais de 1 paradigma de programação, chamadas de multi paradigmas, aumentando a gama de análises e soluções. Os paradigmas existentes, são: Funcional, Lógico, Declarativo, Imperativo, Orientado a Objetos e Orientado a Eventos.

Portanto, os paradigmas de programação servem para o programador analisar qual linguagem será mais eficiente ou confortável para resolver problemas conforme sua necessidade.

**-Questão 9**

A analise semântica, é a terceira fase da compilação, onde se preocupa em analisar o comportamento do programa através da sintaxe da linguagem, antes de executá-lo. É importante saber a semântica para ter conhecimento do que cada instrução faz nessa linguagem. É também onde se verifica os tipos e as variáveis, conforme as regras da linguagem.

**Semantica dinâmica** (Operacional, axiomática, denotacional)**:**

* Operacional: O objetivo é descrever o resultado de um programa através da análise das mudanças na máquina ao executar as instruções. As mudanças no estado da máquina (memória e registradores) definem o significado da instrução.
* Axiomática: Na semântica axiomática, é utilizado um método para provar a exatidão do programa. É feita uma prova, onde cada instrução de um programa tanto é precedida como seguida de uma expressão lógica (asserções) que especifica restrições a variáveis. As que precedem uma instrução são chamadas de pré-condição, e as que seguem uma instrução são as chamadas pós-condições.
* Detonacional: É o método mais abstrato de descrição semântica, porém o mais rigoroso, pois objetos matemáticos são usados para denotar o significado da sintática. Baseia-se em funções recursivas, que serão responsáveis para avaliar as mudanças de estado.