**Nomes:** Alexsander Luis Gomes, Endriu Lucas Graça Ferreira, Pedro Grazziani Fernandes, Rafael Nossal

**Parte I (Tipos de Dados):**

1. Qual(is) a(s) diferença(s) entre uma Linguagem de alto nível e de baixo nível?
   * Linguagens de alto nível são mais fáceis de entender, são ligeiramente mais lentas por conterem abstrações e camadas de código até chegarem ao hardware e são extremamente portáveis (mais usada para escrever softwares multi plataforma e para diversas arquiteturas)
   * Linguagens de baixo nível são mais difíceis de entender, são de alta performance por serem mais próximas da linguagem de máquina (indicada para trabalhar diretamente com hardware), acessa diretamente componentes do hardware (memória e processador, por exemplo), extremamente difícil de entender, requer maior experiência e conhecimento para escrever e não é portável
2. Cite dois ou mais exemplos de Linguagens de alto nível:
   * C#, Java, Python, PHP e Ruby.

C# Language Specification – (https://msdn.microsoft.com/en-us/library/ms228593.aspx)

Java – (http://java.com/en/about/)

Python – (https://www.python.org/about/)

PHP Manual – (https://secure.php.net/manual/en/)

About Ruby – (https://www.ruby-lang.org/en/about/

1. Cite um ou dois exemplos de Linguagens de baixo nível:
   * Assembly.

IBM Knowledge center - (http://www.ibm.com/support/knowledgecenter/SSLTBW\_2.1.0/com.ibm.zos.v2r1.asma400/asmr102112.htm)

1. Cite os tipos de dados primitivos de uma Linguagem voltada ao desenvolvimento web.
   * Inteiro, flutuante, string e booleano.

PHP Manual – (http://php.net/manual/en/language.types.php)

1. Cite os tipos de dados primitivos de uma Linguagem voltada ao desenvolvimento desktop.
   * Boolean, byte, char, short, int, long, float e double.

Oracle Documentation – (http://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html)

1. O que significa uma linguagem com tipagem forte?
   * São as linguagens em que as variáveis devem ter um tipo específico e esse tipo é importante para ela durante a execução.

Artima - (http://www.artima.com/weblogs/viewpost.jsp?thread=7590)

1. O que significa uma linguagem com tipagem fraca?
   * São as linguagens que não se importam com o tipo de dado de uma variável, podendo assim ser de qualquer tipo a qualquer momento e não sendo necessário a conversão de tipo em um determinado momento.

Artima – (http://www.artima.com/weblogs/viewpost.jsp?thread=7590)

**Parte II (Gramática):**

1. Usando a Gramática a seguir, mostre a DEE e a DED (Derivações e Árvores de Derivação) da seguinte instrução: A := B \* (C \* (A + B))

|  |
| --- |
| <atribuição> → <id> := <expr>  <id> → A | B | C  <expr> → <expr> + <termo> | <termo>  <termo> → <termo> \* <fator> | <fator>  <fator> → (<expr>) | <id> |

|  |  |
| --- | --- |
| DEE | DED |
| <id>:=<expr>  <A>:=<expr>  <A>:=<termo>  <A>:=<termo>\*<fator>  <A>:=<fator>\*<fator>  <A>:=<id>\*<fator>  <A>:=B\*<fator>  <A>:=B\*(<expr>)  <A>:=B\*(<termo>)  <A>:=B\*(<termo>\*<fator>)  <A>:=B\*(<id>\*<fator>)  <A>:=B\*(<C>\*<fator>)  <A>:=B\*(<C>\*(<expr>))  <A>:=B\*(<C>\*(<expr>+<termo>))  <A>:=B\*(<C>\*(<termo>+<termo>))  <A>:=B\*(<C>\*(<fator>+<termo>))  <A>:=B\*(<C>\*(<id>+<termo>))  <A>:=B\*(<C>\*(<A>+<termo>))  <A>:=B\*(<C>\*(<A>+<fator>))  <A>:=B\*(<C>\*(<A>+<id>))  <A>:=B\*(<C>\*(<A>+<B>))  ACEITA | <id>:=<expr>  <id>:=<termo>  <id>:=<termo>\*<fator>  <id>:=<termo>\*(<expr>)  <id>:=<termo>\*(<termo>)  <id>:=<termo>\*(<termo>\*<fator>)  <id>:=<termo>\*(<termo>\*(<expr>))  <id>:=<termo>\*(<termo>\*(<expr>+<termo>))  <id>:=<termo>\*(<termo>\*(<expr>+<fator>))  <id>:=<termo>\*(<termo>\*(<expr>+<id>))  <id>:=<termo>\*(<termo>\*(<expr>+<B>))  <id>:=<termo>\*(<termo>\*(<termo>+<B>))  <id>:=<termo>\*(<termo>\*(<fator>+<B>))  <id>:=<termo>\*(<termo>\*(<id>+<B>))  <id>:=<termo>\*(<termo>\*(<A>+<B>))  <id>:=<termo>\*(<fator>\*(<A>+<B>))  <id>:=<termo>\*(<id>\*(<A>+<B>))  <id>:=<termo>\*(<C>\*(<A>+<B>))  <id>:=<fator>\*(<C>\*(<A>+<B>))  <id>:=<id>\*(<C>\*(<A>+<B>))  <id>:=<B>\*(<C>\*(<A>+<B>))  <A>:=<B>\*(<C>\*(<A>+<B>))  ACEITA |
|  |  |

1. A Gramática abaixo é ambígua? Prove:

Utilize a seguinte sentença: a + b + c + a

|  |
| --- |
| <S> → <A>  <A> → <A> + <A> | <id>  <id> → a | b | c |

**R:** A gramática é ambígua pois gera duas árvores de derivação, como mostrado abaixo:

|  |  |
| --- | --- |
| DEE | DED |
| <S>  <A>  <A> + <A>  <id> + <A>  a + <A>  a + <A> + <A>  a + <id> + <A>  a + b + <A>  a + b + <A> + <A>  a + b + <id> + <A>  a + b + c + <A>  a + b + c + <id>  a + b + c + a  ACEITA | <S>  <A>  <A> + <A>  <A> + <id>  <A> + a  <A> + <A> + a  <A> + <id> + a  <A> + c + a  <A> + <A> + c + a  <A> + <id> + c + a  <A> + b + c + a  <id> + b + c + a  a + b + c + a  ACEITA |
|  |  |

1. Descreva, em Linguagem Natural (Português), a Linguagem definida/reconhecida pela seguinte Gramática:

|  |
| --- |
| <S> → <A> <B> <C>  <A> → a<A> | a  <B> → b<B> | b  <C> → c<C> | c |

<S> não terminal deriva em: <A>, <B> e <C> não terminais.

<A> não terminal deriva em: (**a** terminal e <A> não terminal) ou em **a** terminal.

<B> não terminal deriva em: (**b** terminal e <B> não terminal) ou em **b** terminal.

<C> não terminal deriva em: (**c** terminal e <C> não terminal) ou em **c** terminal.

1. Considerando a Gramática a seguir, quais das seguintes sentenças estão na Linguagem gerada por essa Gramática? Apresente todos os testes devidamente identificados.

|  |
| --- |
| <S> → <A>a<B>b  <A> → <A>b | b  <B> → a<B> | a |

* 1. **bbaaaaa**

<S>

<A>a<B>b

<A>ba<B>b

bba<B>b

bbaa<B>b

bbaaa<B>b

bbaaaa<B>b

**bbaaaaab**

**Não aceita**

* 1. **bbbab**

<S>

<A>a<B>b

<A>ba<B>b

<A>bba<B>b

bbba<B>b

**bbbaab**

**Não aceita**

1. Apresente as árvores de derivação e verifique se pertencem ou não a Gramática, para as sentenças **aabbb** e **aaabbb**, a partir da Gramática:

|  |
| --- |
| <S> → a<S>b | ab |

1. **aabbb**

|  |
| --- |
| <S>  a<S>b  aa<S>bb  aaabbb  Não Aceita |
|  |

**b.** **aaabbb**

|  |
| --- |
| <S>  a<S>b  aa<S>bb  aaabbb  Aceita |
|  |