**CANTIDAD DE HOJAS:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | | **UNIVERSIDAD DE BELGRANO** | | |
| **Facultad de Ingeniería y**  **Tecnología Informática**  LENGUAJES DE PROGRAMACIÓN (0050100023LENGP)  Profesor: Mag. Ing. Pablo Pandolfo | | |
| Examen Final diciembre 2020  ALUMNO: LU: FECHA:  CARRERA: | | | | | |
| NOTA: EL EXAMEN ESCRITO ES UN DOCUMENTO DE GRAN IMPORTANCIA PARA LA EVALUACIÓN DE LOS CONOCIMIENTOS ADQUIRIDOS, POR LO TANTO, SE SOLICITA LEER ATENTAMENTE LO SIGUIENTE:   * Responda claramente cada punto, detallando con la mayor precisión posible lo solicitado. * Sea prolijo y ordenado en el desarrollo de los temas. * Sea cuidadoso con las faltas de ortografía y sus oraciones. * No desarrollar el examen en lápiz. * Aprobación del examen: Con nota mayor o igual a 4 (cuatro) * Condiciones de aprobación: 60% correcto. * Duración de examen: 3 horas. | | | | |  |
|  | Ejercicio 1 [3 puntos]: Resuélvase mediante programación lógica la regla: Los números divisibles por 2 y por 3 son divisibles por 6, tener en cuenta los siguientes hechos:  divide(2,6).  divide(2,4).  divide(2,12).  divide(3,6).  divide(3,12).  **divide(6,X) :- divide(2,X), divide(3,X).**  Ejercicio 2 [3 puntos]: Diséñese una GIC en notación BNF para describir la sintaxis de:  En el lenguaje java las declaraciones posibles de variables de los siguientes tipos: int, String, boolean y double. Por ejemplo:  int x, y;  String cadena;  double a;  boolean b;  **<declaraciones> ::= <tipo> <variables>; | <tipo> <variables>; <declaraciones>**  **<tipo> ::= int | String | double | boolean**  **<variables> ::= <var> | <var> , <variables>**  **<var> ::= <letra> | <letra> <letra\_digito>**  **<letra> ::= a | b | c | … | z**  **<letra\_digito> ::= <letra> <letra\_digito> | <digito> <letra\_digito> | <letra> | <digito>**  **<digito> ::= 0 | 1 | 2 | … | 9**  Ejercicio 3 [2 puntos]: Diséñese Diagrama de Sintaxis para las siguientes producciones:  <Sentencias> ::= ( <Asignacion> ;)+  <Asignacion> ::= id = <Expresion>  <Expresion> ::= <Expresion> + <Expresion> | <Expresion> - <Expresion> | id | numero  Ejercicio 4 [2 puntos]: Grafíquese el Árbol de Análisis Sintáctico para la palabra mas corta. Defínase terminales, no terminales.  GIC:  <S> ::= id := <E> | if <E> then <S> | while <E> do <S> | <S>;<S>  <E> ::= <T> or <E> | <T>  <T> ::= <F> and <T> | <F>  <F> ::= not <F> | true | false | ( <E> )  **Palabra mas corta: id := true**  **Terminales = {id, :=, if, then, while, do, ;, or, and, not, true, false, (, ) }**  **No terminales = {<S>, <E>, <T>, <F>}** | |  |