**CANTIDAD DE HOJAS:**

|  |  |
| --- | --- |
|  | **UNIVERSIDAD ARGENTINA DE LA EMPRESA** |
| **Departamento de Tecnología Informática**  TEORÍA DE LA COMPUTACIÓN (3.4.104)  Profesores: Mg. Ing. Pablo Pandolfo /  Ing. Fernando Basteiro |
| Segundo Examen Parcial junio 2022  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: 3 ejercicios bien. * Condiciones de aprobación: 60% correcto. * Condiciones de promoción: 4 o 5 ejercicios bien. * Duración máxima de examen: 3 horas. | |
| **Ejercicio 1 [2 puntos]:** Defínase por comprensión simbólica el lenguaje L que genera la siguiente GIC normalizada en BNF. El alfabeto de los símbolos terminales de la misma es igual a {if, while, do, 1, 2, 3}. El axioma es el símbolo no terminal <START>  <START> ::= <IFs><WHILEs> | <NUMBERS>  <IFs> ::= if<IFs> | if  <WHILEs> ::= while<WHILEs>dodo | while  <NUMBERS> ::= 11<NUMBERS>3 | 11<OTHER\_NUMBERS>3  <OTHER\_NUMBERS> ::= 22<OTHER\_NUMBERS>3 | 2223  Solución:  L1 = {ifn whilem+1 do2m / n >= 1, m >= 0}  L2 = {12n 22m+1 3m+n / n, m >= 1}  L = L1 U L2  **Ejercicio 2 [2 puntos]:** Diséñese una GIC limpia que genere las palabras del siguiente lenguaje L = {s2x wx i2y ty cz h2z / x, y, z >= 0}  Solución:  S -> ABC | AC | AB | BC | A | B | C | λ  A -> ssAw | ssw  B -> iiBt | iit  C -> cChh | chh  **Ejercicio 3 [2 puntos]:** Diséñese un AP por vaciado de pila para el siguiente lenguaje L = {w Ur $r w-1 / w ∈ {a, b}\*, r >= 1}  Solución:  S -> aSa | bSb | U$ | UX$  X -> U$ | UX$  En FNG:  S -> aSA | bSB | UP | UXP  X -> UP | UXP  A -> a  B -> b  P -> $  **Ejercicio 4 [2 puntos]:** Ejemplifíquese el parsing de una cadena de longitud tres que pertenece a un determinado lenguaje formal reconocido por un parser predictivo descendente LL(1), con 5 <= |producciones| <= 10. La gramática NO puede ser regular.  **Ejercicio 5 [2 puntos]:** Ejemplifíquese una MT bicinta con su respectivo programa, que calcule una expresión algebraica con operandos codificados en una determinada base numérica. Al finalizar la computación, la cinta principal debe quedar sin modificaciones y en la segunda cinta debe quedar el resultado de la expresión. Calcular la complejidad temporal de las dos cadenas más cortas. | |