**CANTIDAD DE HOJAS:**

|  | | | **UNIVERSIDAD ARGENTINA DE LA EMPRESA** | | |
| --- | --- | --- | --- | --- | --- |
| **Departamento de Tecnología**  **Informática**  TEORÍA DE LA COMPUTACIÓN (3.4.104)  Profesor: Mag. Ing. Pablo Pandolfo | | |
| Segundo Examen Parcial noviembre 2021  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. * Condiciones de promoción: 70% correcto. * Duración máxima de examen: 3 horas. | | | | |  |
|  |  | Ejercicio 1 [2 puntos]: Defínase por comprensión simbólica el L que genera la GIC:  S -> AB | S’  A -> uAv | uvvv  B -> wBv | wCv  C -> xC | v  S’ -> A’B’ | A’ | B’  A’ -> yA’z | yz  B’ -> zB’&& | z&&  **L = { un vn+2 wm xk  vm+1 /   n, m >= 1, k >= 0} U {yi zi+k &&k / i, k > 0}**  Ejercicio 2 [2 puntos]: Diséñese una GIC en notación BNF para describir la sintaxis de la función map:  map (expresion, Lista de constantes)  Donde expresion  es una expresión de la forma – constante entera (número entero negativo) y Lista de constantes es una lista no vacía de constantes enteras entre corchetes separados por comas. No acepta 08, tampoco -0.  Ejemplos:   map ( -3, [1,20,13,-4] ) , map (-2, [5,2,-3,1] )  Son terminales: map, (), [], símbolo menos, la coma y los dígitos del 0 al 9.  **<S> ::= map  (-<N>,  [<LN>])**  **<N> ::= <N><NS> | <NI>**  **<NS> ::= 0 | <NI>**  **<NI> ::= 1 | … | 9**  **<LN> ::= <N>, <LN> | -<N>, <LN> | 0,<LN> | <N> | -<N> | 0**      Ejercicio 3 [2 puntos]: Dada la GIC LL(1): G = <{S, B, C},  {num, op, (, )}, S, P>, donde S es el axioma, las producciones P:  S -> (B)  B -> num C  C -> op num C | λ  Y la tabla de previsión:   |  | **(** | **)** | **num** | **op** | **$** | | --- | --- | --- | --- | --- | --- | | **S** | S -> (B) | err | err | err | err | | **B** | err | err | B -> num C | err | err | | **C** | err | C -> λ | err | C -> op num C | err |  1. Diséñese el Parser o compilador LL con retroceso (left to right leftmost derivation) en base a la misma. 2. Hágase el análisis sintáctico de la cadena (num op num op num), utilizando además la tabla de previsión, mostrando en cada paso cómo queda la pila, y qué regla o acción utiliza.   a)  Diagrama  Descripción generada automáticamente  b)   | PILA | CADENA | REGLA/ACCION | | --- | --- | --- | | $S | (num op num op num)$ | S -> (B) | | $)B( | (num op num op num)$ | EMP(() | | $)B | num op num op num)$ | B -> num C | | $)Cnum | num op num op num)$ | EMP(num) | | $)C | op num op num)$ | C -> op num C | | $)Cnumop | op num op num)$ | EMP(op) | | $)Cnum | num op num)$ | EMP(num) | | $)C | op num)$ | C -> op num C | | $)Cnumop | op num)$ | EMP(op) | | $)Cnum | num)$ | EMP(num) | | $)C | )$ | C -> λ | | $) | )$ | EMP()) | | $ | $ | **ACEPTAR** |   Ejercicio 4 [2 puntos]: Normalícese la siguiente GIC a FNG y FNC:  S -> AB0 | A0 | 0 | AA | AAA  A -> AB0 | A0 | 0  B -> AB0 | A0 | A1  **FNC:**   1. **A -> B B -> v1 | … | vn => A -> v1 | … | vn** 2. **A -> pBq => A -> PBQ P -> p Q -> q**   **S -> ABC | AC | 0 | AA | AAA**  **A -> ABC | AC | 0**  **B -> ABC | AC | AU**  **C -> 0**  **U -> 1**   1. **A -> BC…Z n > 2 => A -> AX X -> C…Z**   **S -> AX | AC | 0 | AA | AY**  **A -> AX | AC | 0**  **B -> AX | AC | AU**  **C -> 0**  **U -> 1 X -> BC**  **Y -> AA**  **FNG:**   1. **Elim recursion a izquierda**   **A -> AB0 | A0 | 0**  **Entonces:**  **A -> 0 | 0X**  **X -> B0 | 0 | B0X | 0X**   1. **Subo:**   **S -> 0B0 | 0XB0 | 00 | 0X0 | 0 | 0A | 0XA | 0AA | 0XAA**  **A -> 0 | 0X**  **B -> 0B0 | 0XB0 | 00 | 0X0 | 01 | 0X1**  **X -> 0B00 | 0XB00 | 000 | 0X00 | 010 | 0X10 | 0 | 0B00X | 0XB00X | 000X | 0X00X | 010X | 0X10X | 0X**   1. **Reemplazo**   **S -> 0BC | 0XBC | 0C | 0XC | 0 | 0A | 0XA | 0AA | 0XAA**  **B -> 0BC | 0XBC | 0C | 0XC | 0U | 0XU X -> 0BCC | 0XBCC | 0CC | 0XCC | 0UC | 0XUC | 0 | 0BCCX | 0XBCCX | 0CCX | 0XCCX | 0UCX | 0XUCX | 0X**  **C -> 0 U -> 1 A -> 0 | 0X**  Ejercicio 5 [2 puntos]: Diséñese una MT unicinta y una MT bicinta sobre el alfabeto {UADE, UBA} para la función suma f(n, m) = n + m, representando los enteros positivos mediante cadenas de UADE’s. Así, el entero positivo n estaría representado por UADEn. La función se implementa mediante la transformación de UADEnUBAUADEm en UADEn+mUBA. Dejar cabezal/es en el primer símbolo de la cadena/s. Calcúlese complejidad espacial y temporal de ambas MT’s para la cadena UADEUADEUADEUBAUADE.  Gráfico  Descripción generada automáticamente con confianza baja  Gráfico, Gráfico de dispersión  Descripción generada automáticamente   |  | E | | T | | | --- | --- | --- | --- | --- | | Muni | Mbi | Muni | Mbi | | []UADEUADEUADEUBAUADE[] | 7 | 14 | 13 | 26 | | |  |