



PROCESADORES DE LENGUAJE

18 de Junio de 2014

Nombre:

DNI:

Normas para el examen teórico:

- El tiempo máximo para la realización de este ejercicio es de 2 horas.
 - No está permitido entregar ejercicios escritos a lápiz.
 - Los móviles deberán permanecer desconectados durante la realización del ejercicio.
-

1. (1 puntos) **Gestión de la memoria**

Considere la gestión de la memoria de pila en tiempo de ejecución.

- a) Describa el proceso de llamada a una función
- b) Describa el proceso de retorno de una función

2. (1 puntos) **Expresiones regulares**

Dada la siguiente expresión regular:

$$(a^* b (a b)^* a a)^* a^* b (a b)^*$$

obtenga el Autómata Finito Determinista asociado, indicando el conjunto de expresiones regulares punteadas que describen cada estado del autómata.

3. (2 puntos) **Análisis descendente**

Dada la siguiente gramática, hacer las modificaciones necesarias para verificar si cumple las condiciones de LL(1)

S \rightarrow S inst | T R V
T \rightarrow tipo | λ
R \rightarrow blq V fblq | λ
V \rightarrow id S fin | id ; | λ

4. (2 puntos) Análisis ascendente

Comprobar si la siguiente gramática, que genera el lenguaje $\{(xb \mid a^n x b^n), n \geq 0\}$ es SLR(1) o no.

- 1) $S \rightarrow A$
- 2) $S \rightarrow xb$
- 3) $A \rightarrow aAb$
- 4) $A \rightarrow B$
- 5) $B \rightarrow x$

Para ello:

- a) Construye el autómata de transiciones de estados SLR(1)
- b) Construye la tabla de análisis SLR(1)

Estado	a	b	x	\$	S	A	B

5. (2 puntos) ETDS

En un determinado lenguaje se trabaja con ordenes que permiten imprimir páginas individuales y/o rangos de páginas, con la restricción de que no se aceptan páginas individuales después de la aparición del primer rango.

La siguiente gramática modela este tipo de órdenes:

- Imprimir \rightarrow imprime numero MasPaginas
 MasPaginas \rightarrow numero MasPaginas | rango numero MasRangos | λ
 MasRangos \rightarrow numero rango numero MasRangos | λ

Añade a la gramática las acciones semánticas necesarias para sintetizar el atributo cp de **Imprimir** con el resultado de contar la cantidad total de páginas a imprimir. Si en algún rango el número de la primera página es mayor que el de la última, ese rango aporta 0 páginas al total.

Estos son algunos ejemplos:

Entrada	cp
imprime 5	1
imprime 5 7 5	3
imprime 12..15	4
imprime 12..15 3..10 7..5	12
imprime 5 7 5 12..15 3..10	15

6. (2 puntos) **Generación de código**

La siguiente figura muestra la descripción sintáctica del bucle REPEAT-UNTIL en el formato de la herramienta JavaCC.

```
void InstRepeat() :  
{  
  {  
    <REPEAT> <LLAVEAB> ( Instruccion() ) * <LLAVECE>  
    <UNTIL> <PARAB> Condicion() <PARCE> <PYC>  
  }  
}
```

Se pretende enriquecer la gramática anterior para construir el código intermedio de 3 direcciones correspondiente a la instrucción REPEAT. Para ello la función InstRepeat() deberá devolver un objeto Inst cuyo campo code contenga dicho código.

Considere que la función Condicion() devuelve un objeto de la clase Condition con los siguientes campos: code, código intermedio que describe la condición; label_true, etiqueta a la que salta el control en el caso de que la condición sea cierta; label_false, etiqueta a la que salta el control en caso de que la condición sea falsa. Por su parte, la función Instruccion() devuelve un objeto de la clase Inst cuyo campo code contiene el código intermedio asociado a dicha instrucción. Asimismo, se dispone del método getNewLabel(), que devuelve una nueva etiqueta (es decir, una etiqueta no utilizada en ningún punto del código) y del método getNewTemp() que devuelve una referencia a una nueva variable temporal.

Nota: el funcionamiento del bucle REPEAT consiste en ejecutar la lista de instrucciones y a continuación evaluar la condición. En caso de que la condición sea cierta se sale del bucle. Si la condición es falsa se repite el proceso ejecutando la lista de instrucciones y evaluando de nuevo la condición.

[illegible]