



4º Ingeniería Informática

II26 Procesadores de lenguaje

Examen de teoría (16 de septiembre de 2005)

PREGUNTA 1

(5 PUNTOS)

A continuación, se presentan dos posibles extensiones para los lenguajes *Sé* y *Sé-*. Explica claramente qué modificaciones se tendrían que hacer en un compilador de estos lenguajes a Rossi para que las aceptara. Las extensiones son independientes entre sí; no hace falta que consideres sus posibles interacciones.

En tu descripción, procura ser claro, escueto y preciso. En particular, no es necesario que describas partes del compilador que no estén afectadas por las modificaciones. Puedes optar por descripciones algorítmicas o en lenguaje natural para lograr una mayor sencillez en la explicación. También puede facilitarte la exposición una estructura que siga las distintas etapas del compilador.

Explicita cualquier asunción que hagas acerca del compilador o del enunciado propuesto.

Operador signo (2 puntos)

Esta extensión define un nuevo operador: el operador signo. Este operador es prefijo, se escribe mediante la palabra reservada `signo` y está seguido de una expresión de tipo elemental. El resultado depende del valor de la expresión: si ese valor es negativo, el resultado es -1 ; si el valor es cero, también lo es el resultado; en otro caso, el resultado es 1 .

Por ejemplo, la sentencia

```
printf("signo -3= %d\n", signo -3);
```

escribiría `signo -3= -1`.

El operador garantiza que la expresión se evalúa exactamente una vez. En cuanto al tipo del resultado, es siempre entero.

Además de las modificaciones, muestra qué código se generaría para la máquina virtual Rossi con el ejemplo anterior.

Comprobación de ordenación (3 puntos)

Con esta modificación se añade una nueva cabecera, `sorted.h` a la lista de cabeceras que se pueden incluir. El efecto de `sorted.h` es la declaración de la función especial `sorted` que recibe como parámetro una expresión correspondiente a un vector de algún tipo elemental. El resultado de la invocación es cero si el vector no está ordenado de menor a mayor y uno si el vector está ordenado.

Un ejemplo de uso sería:

```
qsort(a);
if ( sorted(a) )
    printf("Bien\n");
else
    printf("Hay un bug en qsort.\n");
```

Este fragmento nos serviría para comprobar si la función `qsort` ordena correctamente el vector que se le pasa como parámetro.

Debes garantizar que el código generado no crece linealmente con el tamaño del vector que se pasa como parámetro.

Además de las modificaciones, muestra qué código se generaría para la máquina virtual Rossi con la sentencia `if` del ejemplo anterior. Supón que `a` es un vector de reales y asígnale la dirección que creas conveniente.

PREGUNTA 2

(2 PUNTOS)

Escribe autómatas finitos deterministas o demuestra que no existen para los siguientes lenguajes:

- Cadenas de cero o más vocales distintas ordenadas alfabéticamente.
- Cadenas de cero o más vocales, posiblemente repetidas, ordenadas alfabéticamente.
- Secuencias de uno o más identificadores separados por comas y ordenados alfabéticamente.
- Números naturales múltiplos de cuatro.

Nota: Para los números naturales, se considera que son válidas secuencias iniciales de ceros.

PREGUNTA 3

(1,5 PUNTOS)

Sea G la siguiente gramática:

$$\begin{aligned} \langle S \rangle &\rightarrow \langle A \rangle \\ \langle A \rangle &\rightarrow \langle A \rangle \langle B \rangle \langle A \rangle \mid \langle B \rangle \langle A \rangle \mid \langle B \rangle \\ \langle B \rangle &\rightarrow \text{num} \mid \langle B \rangle \text{num} \mid \lambda \end{aligned}$$

Supongamos que la primera regla tiene asociada la acción

$$\langle S \rangle \rightarrow \langle A \rangle \{ \text{si } \langle A \rangle.\text{nrc} \text{ entonces escribe ("No hay problemas")} \text{ si no escribe ("Hay repetidos consecutivos")} \}$$

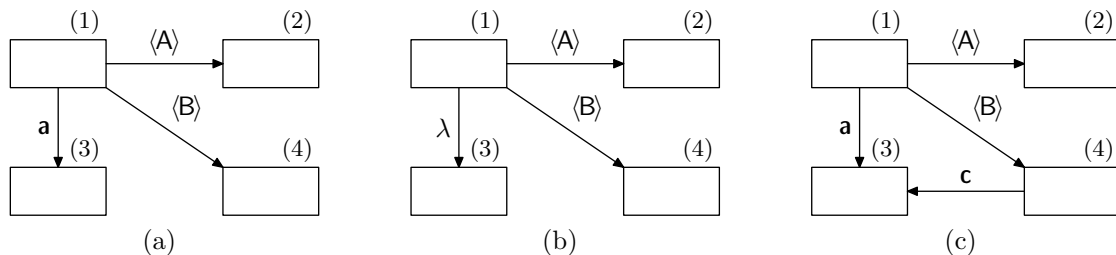
donde *nrc* ("no repetidos consecutivos") es un atributo de tipo lógico que es cierto si la secuencia de números enteros es vacía o no hay dos números *consecutivos* iguales y falso en caso contrario. Añade, *al final de cada una de las restantes reglas*, las acciones semánticas necesarias para que se calcule el valor de este atributo.

Puedes utilizar los atributos adicionales que consideres necesarios, pero ninguna variable global. Además, los atributos que añadas deben ser de tipo entero o lógico. Explica el significado de cada atributo que añadas. Asume que los componentes de la categoría **num** tienen su valor en el atributo v .

PREGUNTA 4

(1,5 PUNTOS)

Para cada uno de los siguientes autómatas



escribe una gramática SLR cuyo autómata de prefijos viables se corresponda con él o demuestra que no es posible.

Duración del examen: 4 horas
¡Buena suerte!