

PRACTICA 10: Caracterización de un lenguaje independiente del contexto

10.1. Introducción

Un lenguaje independiente del contexto puede ser finito o infinito. Dada una gramática independiente del contexto, $G \equiv (V, \Sigma, S, P)$, el siguiente algoritmo permite determinar si $L(G)$ es finito o infinito.

1. Si la gramática no está en forma normal de Chomsky, escribirla en forma normal. Sea $G' \equiv (V', \Sigma, S, P')$ la gramática escrita en forma normal de Chomsky.
2. Crear un grafo dirigido, en el que el número de nodos será $|V'|$ y habrá un nodo por cada símbolo no terminal de V' .
3. Para cada producción de P' de la forma $A \rightarrow BC$, incluir en el grafo dos arcos dirigidos: de A a B y de A a C .
4. $L(G)$ es finito si en el grafo no existen ciclos dirigidos.
5. $L(G)$ es infinito si hay algún ciclo dirigido en el grafo.

Consideremos la siguiente gramática:

$S \rightarrow AB$
 $A \rightarrow BC \mid a$
 $B \rightarrow CC \mid b$
 $C \rightarrow a \mid AB$

La gramática está escrita en forma normal de Chomsky. En caso que no lo estuviera, habría que reescribirla en forma normal. Si construimos el grafo dirigido correspondiente a esta gramática, obtenemos el que aparece en la Figura 10.1. Puesto que en ese grafo aparecen ciclos dirigidos (por ejemplo $A \rightarrow B \rightarrow C \rightarrow A$), concluimos que $L(G)$ es infinito.

También es posible determinar si el lenguaje $L(G)$ que genera una gramática es vacío o no. Basta para ello con aplicar la primera etapa del algoritmo de eliminación de símbolos y producciones inútiles que se muestra en la figura 10.2.

Si el símbolo de arranque de la gramática, S , resulta eliminado en ese algoritmo es porque no existe ninguna derivación $S \Rightarrow^* w$ que a partir de S derive en una cadena w que contenga sólo símbolos terminales. Si dicha derivación no existe ello implica que $L(G)$ es vacío.

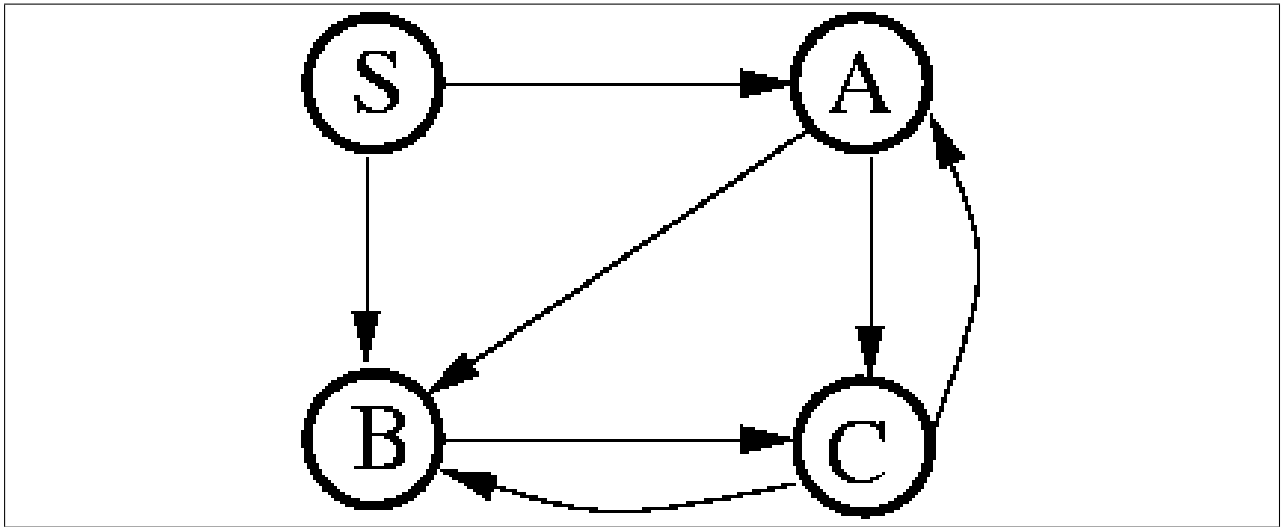


Figura 10.1: Grafo dirigido correspondiente a la gramática.

10.2. Práctica

La práctica consiste en realizar un programa en C++ que tome como entrada un fichero en el que se especifique una gramática independiente del contexto, G y que imprima como salida solamente una de las tres cadenas “finito vacío”, “finito no vacío” o “infinito”

- $V' = \emptyset$
- **for** cada producción de la forma $A \rightarrow \omega$ **do**
- $V' = V' \cup \{A\}$
- **while** (V' cambie) **do**
- **for** cada producción de la forma $B \rightarrow \alpha$ **do**
- **if** (todos los símbolos no-terminales de α pertenecen a V')
- $V' = V' \cup \{B\}$
- **Eliminar** todas las variables ($C \in V$) que estén en V y no en V'
- **Eliminar** todas las producciones donde aparezca una variable de las eliminadas en el paso anterior

Figura 10.2: Primera etapa del algoritmo de eliminación de símbolos y producciones inútiles