

- 1) **Una gramática es un conjunto de reglas para definir un conjunto finito de sentencias validas en un lenguaje.**
FALSO. Es un conjunto de reglas finito para definir un conjunto infinito.
- 2) **La ausencia de la declaración de una variable en una unidad de código no siempre produce un error semántico.**
VERDADERA. Hay que declarar siempre la variable.
- 3) **Un lenguaje es expresivo si todas las variables declaradas en él tienen un tipo definido en compilación.**
FALSO.
- 4) **Una variable es estática en cuanto a su l-valor si se aloca en memoria en el mismo momento que la unidad que la contienen y se desaloca cuando la unidad termina su ejecución.**
FALSA. Es automática.
- 5) **Se puede decir que en los lenguajes analizados el concepto de “variable local” infiere el tiempo de vida y no el alcance.**
FALSA. Infiere en el alcance que es limitado y el T.V se define dentro del bloque.
- 6) **Un metalenguaje es un lenguaje que se utiliza solamente para definir la sintaxis de sentencias de lenguajes de programación.**
VERDADERO. Entra en la definición de BNF
- 7) **Phyton es un lenguaje puramente orientado a objetos.**
FALSO. Es orientado a objetos, pero también tienen programación imperativo y funcional.
- 8) **Un lenguaje es legible cuando permite definir nuevos tipos de datos.**
FALSO. Es legible cuando se escribe correctamente, con comentarios, sin ambigüedades. Cuanto mas formal y preciso sea la definición.
- 9) **Cuando se compila un programa solo se detectan errores sintácticos en cambio cuando se interpreta se detectan errores semánticos y sintácticos.**
FALSO. Hace las dos cosas también.
- 10) **En pascal una variable declarada en la zona de declaraciones del bloque de programa es estática en cuanto a su l-valor, pero es dinámica en cuando a su r-valor**
FALSA. Es automática en cuanto a l-valor.
- 11) **En C, si asignamos a una variable puntero, la dirección de otra variable, estamos seguros de que la variable puntero nunca contendrá referencias sueltas.**
FALSA.
- 12) **Si un lenguaje provee encapsulamiento de código entonces posee todo lo necesario para definir tipos abstractos de datos.**
FALSA.
- 13) **Un lenguaje es fuertemente tipado si asegura que no hay conflictos de tipos en ejecución.**
VERDADERA.
- 14) **Un tipo de datos es un atributo que indica a una variable solamente el conjunto de valores y operaciones posibles que la misma puede tener.**
VERDADERA.
- 15) **Un lenguaje puede ser evaluada por las facilidades que brinda para escribir sentencias en él.**
VERDADERA. Si ya que brinda expresividad.
- 16) **La semántica de una expresión en un lenguaje de programación estable los posibles valores que la misma puede representar.**
VERDADERA. La semántica se encarga de esto.
- 17) **El link estático de una unidad de programa dentro del registro de activación de un lenguaje basado en pila define cual es la unidad de programa que la invoco.**
FALSA. Link estático define quien lo contiene. Link dinámico quien lo invoco.
- 18) **El concepto de claridad de bindings esta asociado a la confiabilidad que brinda el lenguaje al momento en que se define el atributo de la entidad.**
FALSO. La confiabilidad esta relacionada con la seguridad, chequeo de tipos y manejo de excepciones.
- 19) **Una de las ventajas de la interpretación respecto a la compilación esta relacionada con la puesta a punto de los programas.**
VERDADERA.