Pregunta 1

Definir el concepto de lenguaje de proposiciones.

Si σ un conjunto finito de átomos, definimos el lenguaje de las σ-fórmulas proposicionales como el conjunto de elementos generados por las siguientes reglas:

(1) Todo elemento de σ es una fórmula.

(2) Si φ es una fórmula, $\neg\varphi$ también lo es.

(3) Si φ, ψ son fórmulas, entonces $(\varphi \vee \psi)$, $(\varphi \wedge \psi)$, $(\varphi \rightarrow \psi)$ y $(\varphi \leftrightarrow \psi)$ son también fórmulas.

Pregunta 2

Definir el concepto de interpretación y los conceptos de tautología, fórmula satisfactible y contradicción.

Si σ un conjunto finito de átomos, una σ-interpretación es una asignación de los valores de verdad a los elementos de σ.

Una fórmula es una tautología, si es cierta en todas las interpretaciones.

Una fórmula es satisfactible, si es cierta en alguna interpretación.

Una fórmula es una contradicción, si es falsa en todas las interpretaciones.

Pregunta 3

Definir el concepto de fórmulas equivalentes.

Dos fórmulas φ, ψ son lógicamente equivalentes, si para toda interpretación I, tenemos que $I(\varphi) = I(\psi)$. Es decir, dos fórmulas son lógicamente equivalentes, si no hay ninguna interpretación que las distinga.

Pregunta 4

Definir los conceptos de literal, cláusula y fórmula en forma normal conjuntiva.

Un literal es un átomo o a la negación de un átomo.

Una cláusula es una disyunción de literales.

Una fórmula φ está en forma normal conjuntiva, si φ es de la forma $\varphi_1 \land \ldots \land \varphi_n$ donde $\varphi_1, \ldots, \varphi_n$ son cláusulas.

Pregunta 5

Explicar el algoritmo visto en clase para transformar una fórmula proposicional en una fórmula equivalente en forma normal conjuntiva.

Hay que seguir el siguiente proceso:

(1) Aplicar las reglas:
$\varphi \leftrightarrow \psi \equiv (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$,
$\varphi \rightarrow \psi \equiv \neg\varphi \vee \psi$.

(2) Aplicar las reglas:
$\neg(\neg\varphi) \equiv \varphi$,
$\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi$
$\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi$.

(3) Aplicar las reglas:
$\varphi \vee (\psi \wedge \chi) \equiv (\varphi \vee \psi) \wedge (\varphi \vee \chi)$,
$\varphi \wedge (\psi \vee \chi) \equiv (\varphi \wedge \psi) \vee (\varphi \wedge \chi)$.

Pregunta 6

Explicar en qué consiste el problema SAT y explicar el interés de dicho problema.

El problema SAT consiste en determinar si una fórmula proposicional en forma normal conjuntiva es satisfactible.

El interés del problema SAT radica en que hay muchos problemas prácticos que se pueden representar mediante una fórmula proposicional, de manera que el hecho de que el problema tenga solución significa que la fórmula asociada es satisfactible.

Pregunta 7

Explicar qué es un SAT-solver y dar ejemplos de problemas que se pueden resolver mediante SAT-solvers.

Un SAT-solver es un programa que resuelve el problema SAT.

Entre los problemas que se pueden resolver mediante SAT-solvers, tenemos los siguientes:

(1) El problema de colorear un mapa de un continente con cuatro colores de manera que no haya dos países vecinos que tengan el mismo color.

(2) El problema de instalar paquetes de actualización en un ordenador satisfaciendo una serie de restricciones.

(3) El problema de formar un equipo de personas lo más pequeño posible para realizar una serie de tareas.

(4) Problemas para confeccionar horarios de hospitales, escuelas o líneas aéreas.

(5) La resolución de sudokus.

Pregunta 8

Explicar el interés del método de Davis-Putnam.

Es el método en el que está basado el diseño de los SAT-solvers. Es un método mucho más eficiente que el algoritmo consistente en generar todas las interpretaciones posibles y ver si alguna de ellas satisface la fórmula de la entrada.

El interés del método también radica en el hecho de que las reglas de Davis-Putnam preservan la satisfactibilidad de la fórmula de la entrada.

Pregunta 9

Definir el concepto de resolvente de dos fórmulas proposicionales.

Para dos cláusulas proposicionales φ_1, φ_2, si existe un literal ψ_1 en φ_1 que es complementario de un literal ψ_2 en φ_2, se suprimen ψ_1, ψ_2 de φ_1, φ_2 respectivamente, y se construye la disyunción de las cláusulas resultantes. La cláusula así construida se llama resolvente de φ_1 y φ_2.

Pregunta 10

Explicar en qué consiste el algoritmo de resolución para lenguajes de proposiciones.

El algoritmo está basado en el teorema de resolución y se utiliza para diseñar demostradores. Recibe como entrada una lista de fórmulas $\varphi_1, \ldots, \varphi_n, \varphi$, y da como salida "éxito" si φ es consecuencia lógica de $\varphi_1, \ldots, \varphi_n$ y da "fallo" en caso contrario.

El algoritmo consta de las siguientes etapas:

(1) Se calcula una forma normal conjuntiva $\psi_1 \wedge \ldots \wedge \psi_k$ de la fórmula $\varphi_1 \wedge \ldots \wedge \varphi_n \wedge \neg\varphi$

(2) El algoritmo va calculando resolventes a partir de las fórmulas $\psi_1, \ldots, \psi_k$ hasta que se obtenga la cláusula vacía $\square$ o hasta que no se puedan computar más resolventes.

El algoritmo dará entonces "éxito" si se obtiene la cláusula vacía, y dará "fallo" en caso contrario.