IIC2233

fblazco

August 15, 2024

1 Clase 1

1. **Fecha Evaluaciones**

2 Clase 3

2.1 Induccion Estructural

Clase anterior: Teorema

1. Principio del buen orden
2. Principio de induccion simple
3. Principio de induccion fuerte

Definiciones Inductivas: Para definir inductivamente un conjunto necesitamos

1. Un conjunto de elementos base no necesariamente finito
2. Un conjunto finito de reglas de construccion de nuevos elementos del conjunto a partir de elementos que ya estan en elementos
3. Establecer que el conjunto es el menor que cumple las reglas

Ejemplo: El conjunto de numeros pares es el menor conjunto tq

1. El 0 siempre es pares
2. Si n es un numero par, n+2 es numero par

Definicion (listas enlazadas) $\mathcal{L}$n: El conjunto $\mathcal{L}$n es el menor conjunto que cumple con las sgts. reglas

1. $\phi \, \epsilon \, \mathcal{L}$n
2. Si L ϵ $\mathcal{L}$n y k ε N, entonces L $\rightarrow$ k ϵ Ln

Cuando dos listas enlazadas son iguales?

1. Si alguna es ϕ, son iguales ssi la otra tmb es vacia
2. Si ninguna es vacia entonces estamos en un escenario

$$L_1 \rightarrow k_1 \text{ vs } L_2 \rightarrow k_2$$

en este caso, resulta natural considerar

$$L_1 \rightarrow k_1 = L_2 \rightarrow k_2 \text{ si solo si } L_1{=}L_2 \text{ y } k_1{=}k_2$$

2.2 Principio de Induccion Estructural

Sea A un conjunto definido inductivamente y P una propiedad sobre los elementos de A. Si se cumple que:

1. Todos los elementos base de A cumplen la propiedad P
2. Para cada regla de construccion, si la regla se aplica sobre elementos en A que cumplen la propiedad P

Ejemplo P(L): L tiene el mismo numero de flechas que de elementos

BI: El unico caso base de es la lista vacia
HI:

(terminar demostracion)
Para demostrar propiedades en Ln definiremos mas operadores(agregar operadores)

1. Largo, recibe lista y enetrega numero de elementos
2. Suma, recibe lista y entrega la suma de sus elementos
3. Maximo, recibe lista y entrega el maximo, o -1 si es vacia

$$\text{max: } L_n \rightarrow \text{N} \vee \{\text{-1}\}$$

4. Cabeza, recibe una lista no vacia y entrega su primer elemento

$$\text{head: } L_n \;/\; \{\phi\} \rightarrow L_n$$

Si k ϵ N, entonces suf($\rightarrow$k) = ϕ
Si

Teorema (props, listas): Si L, L_1, L_2 ϵ L_n, entonces:

1. sum(L) >= 0
2. max(L) <= sum(L)

Teorema (prop. 4 de listas) Sean, L_1, L_2 ϵ L_n. Si L_1, $L_2 \neq \phi$ (terminar)
Demostracion: Sea

$$L_1 = \!\rightarrow K_1$$
$$L_2 = \!\rightarrow K_2$$
$$\text{tq: suf}(L_1) = \text{suf } (L_2)$$
$$\text{sum}(L_1) = \text{sum}(L_2)$$

Por definicion de igualdad de listas

$$L_1{=}L_2$$
$$\phi \rightarrow \text{k} = \phi \rightarrow \text{y}$$
$$\phi = \phi$$
$$\text{y k= j}$$

3 Clase 4

Cambios Evaluaciones:

T1: 02 de Septiembre 17:30
T2: Entrega 04 de Septiembre

3.1 Logica Proposicional

Sintaxis: Sea P un conjunto de variables Proposicionales

1. Si p ϵ P, entonces pϵ $\mathcal{L}$(P)
2. Si ϕ ϵ $\mathcal{L}$(P), entonces $\neg\phi\epsilon$ $\mathcal{L}$(P)
3. Si ϕ,ψ ϵ $\mathcal{L}$(P) y ϵ $(\wedge, \vee, \rightarrow, \leftrightarrow)$, entonces (ψxϕ) ϵ $\mathcal{L}$(P)