Maestría en Computación

Instituto de Postgrados - Facultad de Ingeniería Electrónica y Telecomunicaciones

Universidad del Cauca

Prueba Inducción y Lógica

Profesor: Julio Ariel Hurtado Alegría

Cuestionario

(Tiempo 2 Horas)

1. (1.0 puntos) Dada la sucesión:

$$t_1 = 1$$

$$t_{n+1} = 2t_n + 1$$

Demuestre usando inducción, que $t_n \leq 2^n$

($Ayuda$: $considere\ que\ \ 2^n - 1 \leq 2^n\ si\ requiere\ de\ una\ hipótesis\ más\ fuerte$

$que\ simplifique\ la\ demostración$)

2. (1.0 punto) Sea $P = \{p\}$ el conjunto de proposiciones. Se define la función $largo(\varphi)$ la cuál es igual al número de símbolos que se usan para escribir una fórmula proposicional $\varphi \in L(P)$. Sea $M = \{n \in \mathbb{N} \mid Existe\ \beta \in L(P)\ tal\ que\ largo(\beta) = n\}$, es decir M corresponde al conjunto de todos los posibles largos de las fórmulas proposicionales en $L(P)$. Usando inducción, demuestre que $M = \{1,4,5\} \cup \{n \in \mathbb{N} \mid n \geq 7\}$.

Ejemplo: $largo(p\ ) = 1$, $largo((\neg p))$ = 4, $largo((p\ v\ q))$ = 5, $largo((\neg\psi))$ = 3+ $largo(\psi)$

$Ayuda$: $su\ demostración\ debe\ incluir\ además\ las\ razones\ por\ las\ que$ 2, 3 y 6 no $pertenecen\ a\ M$

3. Sea $\mathcal{L}_{\mathbb{N}}$ el dominio de las listas de números naturales. Sea $L \in \mathcal{L}_{\mathbb{N}}$, $|L|$ denota el largo de la lista L.

   a. (1.0 punto) Defina inductivamente las funciones sufijo y prefijo extendido $SufE: \mathcal{L}_{\mathbb{N}}\ x\ \mathbb{N} \rightarrow \mathcal{L}_{\mathbb{N}}$ y $PrefE: \mathcal{L}_{\mathbb{N}}\ x\ \mathbb{N} \rightarrow \mathcal{L}_{\mathbb{N}}$ tales que, dada una lista L y un número natural n, $SufE(L, n)$ entrega la lista sufija de L de largo n, y $PrefE(L, n)$ entrega la lista prefija de L de largo n. Por ejemplo sus funciones deben ser tales que:

SufE(→ 3 → 4 → 7 → 1 → 9, 3) = → 7 → 1 → 9

SufE(→ 3 → 4 → 7 → 1 → 9, 5) = → 3 → 4 → 7 → 1 → 9
PrefE(→ 3 → 4 → 7 → 1 → 9, 2) = → 3 → 4
PrefE(→ 3 → 4 → 7 → 1 → 9, 10) = → 3 → 4 → 7 → 1 → 9

b. (1.0 punto) Defina inductivamente la función $Ocurre: \mathcal{L}_{\mathbb{N}}\, x\, \mathcal{L}_{\mathbb{N}} \rightarrow \mathbb{N}$ que dadas dos listas $L1\ y\ L2$ con $L1 \neq \emptyset$, entrega la cantidad de veces que $L1$ se repite en $L2$. Por ejemplo dadas las listas:

L1 = → 2 → 3 → 2

L2 = → 2 → 3 → 2 → 2

L3 = → 1 → 2 → 3 → 2 → 3 → 2 → 5 → 2 → 3 → 2

Se tiene que: $Ocurre(L1, L3) = 3\ y\ Ocurre(L1, L2) = 1$

c. (1.0 punto) De una implementación en Scheme de la función $Ocurre$ y de las funciones auxiliares que haya considerado en la solución del punto b.

Considere las siguientes funciones de Scheme:

- *(null? lst)* devuelve *#t* si lst = *'() ,* y *#f* en caso contrario
- *(reverse lst)* devuelve *lst* invertida
- *(append lstA lstB)* devuelve la lista *lstA* concatenada con la lista *lstB* (ambas deben ser listas).
- *(list elem1 elem2 elem3…)* devuelve la lista *(elem1 elem2 elem3)*. Por ejemplo *(list 'a 'b)* devuelve la lista *(a b)*.

"Las ciencias de la computación están tan relacionadas

con las computadoras como la astronomía con los telescopios"

Edsger Dijkstra