ISIS 1105 Diseño y Análisis de Algoritmos
Semestre 2011-10 – Parcial 1
Marzo 1, 2011
Prof. Rodrigo Cardoso

1 ***[30/100]***

Se sabe que las sucesiones a y b cumplen

```
[r1]   a(n+1) = a(n) + b(n)     , n≥0
[r2]   b(n+1) = 3b(n) - a(n)    , n≥0
[r3]   a[1] = 1
[r4]   a[2] = 3
```

1a [20/30] Encuentre soluciones cerradas para `a(n)` y `b(n)`.

Por `[r1]`:

[r5] $b(n) = a(n+1) - a(n) \quad , n\geq 0$

$\Rightarrow$

[r6] $b(n+1) = a(n+2) - a(n+1), n\geq -1$

$\Rightarrow$ $\langle$ Remplazo [r5] y [r6] en [r2] $\rangle$

$a(n+2) - a(n+1) = 3(a(n+1) - a(n)) - a(n), n\geq 0$

$\Rightarrow$

$a(n+2) - 4a(n+1) + 4a(n) = 0, n\geq 0$

$\equiv$

$(E^2 - 4E + 4)a = 0$

$\equiv$

$(E - 2)^2 a = 0$

$\Rightarrow$ Existen A,B constantes, tales que:

[r7] $a(n) = A*2^n + B*n*2^n , n\geq 0$

$\Rightarrow$ $\langle$ [r7],[r3], r[4] $\rangle$

$a(1) = 1 = 2A + 2B$

$a(2) = 3 = 4A + 8B$

$\Rightarrow$

A=1/4

B=1/4

$\Rightarrow$

[r8] $a(n) = 2^{n-2}(1+n) \quad , n\geq 0$

[10/30]

$\Rightarrow$ $\langle$Remplazando [r8] en [r5]$\rangle$

$b(n) = 2^{n-1}(2+n) - 2^{n-2}(1+n) \quad , n\geq 0$

$\Rightarrow$

[r9] $b(n) = 2^{n-2}(3+n) \quad , n\geq 0$

[10/30]

1b [10/30] Pruebe o refute:

[1b1] $a = O(b)$

[1b2] $a = \theta(b)$

Se calcula

$$\lim_{n\to\infty} \frac{a(n)}{b(n)} = \lim_{n\to\infty} \frac{2^{n-2}(1+n)}{2^{n-2}(3+n)} = 1$$

Entonces, ya que el límite existe y es mayor que `0`:

[1b1] es verdadero

[5/30]

[1b2] es verdadero

[5/30]

2 ***[30/100]***

Suponga que GCL se aumenta con una instrucción

$$S1\ \langle B\rangle\ S2$$

cuya semántica operacional sea la secuenciación condicionada de S1 y S2, dependiendo de B. Más exactamente, ejecutar S1 y si B es verdadero después de terminar normalmente S1, ejecutar S2.

2a [15/30] Simule S1 ⟨B⟩ S2 con un programa GCL que no utilice esta instrucción.

```
S1 ⟨B⟩ S2   ≈   S1; if B → S2 [] ¬B → skip fi
```

[15/30]

2b [15/30] Enriquezca el cálculo de Hoare para GCL con una regla de inferencia que permita concluir la corrección de afirmaciones como

$$\{Q\}\ S1\ \langle B\rangle\ S2\ \{R\}$$

Una regla de inferencia adecuada sería, de acuerdo con **2a**:

$$\text{SECC:}\quad \frac{\{Q\}\ S1\ \{R1\},\ \{R1 \wedge B\}\ S2\ \{R\},\quad R1 \wedge \neg B \Rightarrow R}{\{Q\}\ S1\ \langle B\rangle\ S2\ \{R\}}$$

[15/15]

3 ***[40/100]***

Sea a[0..n]:**int**, un arreglo de enteros que guarda los coeficientes de un polinomio A(x) de grado n, de modo que

$$A(x) = a[n]*x^n + a[n-1]*x^{n-1} + \ldots + a[1]*x + a[0].$$

Si para un intervalo r..s, r<s, se sabe que $A[r]\le 0$ y $A[s]>0$, se puede afirmar que, para algún x, $r\le x<s$, $A[x]=0$. El siguiente algoritmo determina un intervalo i..i+1 que contiene un tal x, con $r\le i<s$.

```
[Ctx C:   a:[0..n]:int ∧ (∀x|: A(x) = (+k| 0≤k≤n: a[k]*x^k))

 {Pre Q: A[r]≤0<A[s]}

 i,j:= r,s;

 {Inv  P: r≤i<j≤s ∧ A[i]≤0<A[j] }

 do i+1≠j   →     d:= (i+j)÷2;
                  h,ad:= n+1,0;

                  {Inv P1: P ∧ ad = (+k| h≤k≤n : a[k]*d^(k-h)) ∧ 0≤j≤n}

                  do h≠0 →    h:= h-1;
                              ad:= ad*d + a[h]
                  od;

                  {R1: P ∧ ad = (+k| 0≤k≤n : a[k]*d^k)}
```

```
                        if ad≤0 →   i:= d
                        [] ad>0 →   j:= d
                        fi

    od

   {Q1: r≤i≤s ∧ A[i]≤0<A[i+1] }
   {R : (∃x| i≤x<i+1: A(x)=0)}
   ⌋
```

3a **[20/20]** Enuncie todas las *obligaciones de prueba*, i.e., las condiciones que deben cumplirse para garantizar la corrección del algoritmo, con las anotaciones dadas.

Contexto

(0) El contexto `C` se mantiene porque `a` nunca es cambiado. No se considerará el contexto al enunciar las demás obligaciones, por simplicidad.

Corrección del ciclo externo

(1) `P` vale antes:

```
{Q} i,j:= r,s {P}
```

(2) `P` sirve:

```
P ∧ i+1=j ⇒ Q1
```

(2a) `Q1 ⇒ R`

(3) `P` invariante:

```
{P ∧ i+1≠j} ⟨cuerpo de ciclo externo⟩ {P}
```

(4) El ciclo externo termina

Corrección del ciclo interno

(5) `P1` vale antes:

```
{P1 ∧ i≠m}
d:= (i+j)÷2;
j,r:= n+1,0;
{P1}
```

(6) `P1` sirve:

```
P1 ∧ j=0  ⇒  R1
```

(7) `P1` invariante:

```
{P1 ∧ j≠0}
j:= j-1;
r:= r*d + a[j]
{P1}
```

(8) El ciclo interno termina.

[20/20]

[Cada obligación: +2. No se cobra no mencionar el contexto en la obligaciones 1..8]

3b **[5/5]** Justifique la terminación del algoritmo, explicando por qué terminan los dos ciclos.

El ciclo interno termina porque es un ciclo lineal (**for**) que cuenta en forma regresiva desde `n+1` hasta `0`, con la variable h.

El ciclo externo termina porque, en cada iteración, el intervalo de búsqueda se reduce a la mitad.

[5/5]

3c **[15/15]** Estime `T(r,s,n)` la complejidad temporal del algoritmo, como θ(`f(r,s,n)`). Considere como operaciones básicas las multiplicaciones. Una multiplicación cuesta `1`.

El ciclo externo se realiza `log (s-r+1)` veces (en realidad, ⌈`log (s-r+1)`⌉), ya que, como se anotó en **3b**, el algoritmo parte el intervalo de búsqueda en la mitad, en cada iteración. Empieza con un intervalo para `i` de tamaño `s-r+1` y termina con un intervalo de tamaño `1`.

El ciclo interno tiene n multiplicaciones en el cálculo de `A(d)`.

Resumiendo:

```
T(r,s,n)  = θ((+u| 1 ≤ u ≤ log(s-r+1) : n))
          = θ(n * log(s-r+1))
```

[15/15]