Pauta de Corrección

Primer Certamen
Algoritmos y Complejidad

27 de octubre de 2018

1. Ambas partes son independientes.

   a) Bajo el supuesto indicado, podemos plantear la ecuación para la aproximación x^+:

$$\begin{aligned}
\frac{x_{n+2} - x^+}{x_{n+1} - x^+} &= \frac{x_{n+1} - x^+}{x_n - x^+} \\
(x_{n+2} - x^+)(x_n - x^+) &= (x_{n+1} - x^+)^2 \\
x_{n+2}x_n - (x_{n+2} + x_n)x^+ + (x^+)^2 &= x_{n+1}^2 - 2x_{n+1}x^+ + (x^+)^2 \\
x_{n+2}x_n - x_{n+1}^2 &= (x_{n+2} - 2x_{n+1} + x_n)x^+
\end{aligned}$$

   Despejando:

$$x^+ = \frac{x_{n+2}x_n - x_{n+1}^2}{x_{n+2} - 2x_{n+1} + x_n}$$

   Esta fórmula es muy inestable numéricamente (el resultado es dividir dos expresiones que restan cantidades muy parecidas), puede escribirse en forma más estable como:

$$x^+ = x_n - \frac{(\Delta x_n)^2}{\Delta^2 x_n}$$

   donde:

$$\begin{aligned}
\Delta x_n &= x_{n+1} - x_n \\
\Delta^2 x_n &= \Delta(\Delta x_n) \\
&= \Delta(x_{n+1} - x_n) \\
&= x_{n+2} - 2x_{n+1} + x_n
\end{aligned}$$

   Por esta fórmula se le conoce como el *proceso* Δ^2 *de Aitken*, nombrado por Alexander Aitken quien lo introdujo en 1926.

   b) Sea cual sea el resultado de la primera parte de la pregunta, resulta el algoritmo 1 para aproximar el punto fijo de $g(x)$, con el punto de partida x_0, (reemplace la fórmula para x^+ a asignar a x_0).

Algoritmo 1: Algoritmo usando aceleración de Aitken

function Aitken(g, x_0, ϵ)
  repeat
    $x_1 \leftarrow g(x_0)$
    $x_2 \leftarrow g(x_1)$
    $x_0 \leftarrow x_0 - (\Delta x_0)^2 / \Delta^2 x_0$
  until $|x_2 - x_0| \leq \epsilon$
  return x_0

Puntajes

Total		30
a) Derivar la aproximación x^+	20	
b) Algoritmo claro	10	

2. Buscamos una fórmula gaussiana exacta para polinomios hasta grado 3, por lo que requeriremos $n = 2$ puntos. La teoría nos indica buscar una secuencia de polinomios ortogonales con función de peso $w(x) = x$ sobre $[0, 1]$. La fórmula será del tipo:

$$\int_0^1 x f(x)\,\mathrm{d}x \approx A_0 f(x_0) + A_1 f(x_1)$$

Usamos el proceso de Gram-Schmidt para construir una base ortogonal, partiendo con los vectores $1, x, x^2$. El producto interno relevante es:

$$\langle f, g\rangle = \int_0^1 w(x) f(x) g(x)\,\mathrm{d}x$$

$$\begin{aligned}
p_0(x) &= 1 \\
p_1(x) &= x - \frac{\langle x, p_0(x)\rangle}{\langle p_0(x), p_0(x)\rangle} p_0(x) \\
p_2(x) &= x^2 - \frac{\langle x^2, p_1(x)\rangle}{\langle p_1(x), p_1(x)\rangle} p_1(x) - \frac{\langle x^2, p_0(x)\rangle}{\langle p_0(x), p_0(x)\rangle} p_0(x)
\end{aligned}$$

Nos interesan los ceros de p_2, llamémosles x_0 y x_1.

Si llamamos $\ell_i(x)$ a los polinomios de Lagrange:

$$\ell_i(x) = \prod_{\substack{0 \le k \le n \\ k \ne i}} \frac{x - x_k}{x_i - x_k}$$

los coeficientes están dados por:

$$A_i = \int_0^1 x \ell_i(x)\,\mathrm{d}x$$

Un amable sistema de álgebra simbólica (Maxima, ver el archivo `p2.mc` adjunto) entrega:

$$\begin{aligned}
p_0(x) &= 1 \\
p_1(x) &= x - \frac{2}{3} \\
p_2(x) &= x^2 - \frac{6}{5}x + \frac{3}{10}
\end{aligned}$$

Los ceros de p_2 son:

$$x_0 = \frac{6 - \sqrt{6}}{10} \qquad x_1 = \frac{6 + \sqrt{6}}{10}$$

Con esto los coeficientes buscados son:

$$A_0 = \frac{9 + \sqrt{6}}{36} \qquad A_1 = \frac{9 - \sqrt{6}}{36}$$

Puntajes

Total		30
Planteo como integración gaussiana de 2 puntos	10	
Explicar cómo obtener $p_2(x)$	10	
Nodos son los ceros de p_2	4	
Fórmulas para coeficientes	6	

3. Por turno.

   a) La idea deberá ser:

      - Mover los $n-1$ platos superiores de A a C
      - Mover el plato mayor de A a B
      - Mover los $n-1$ platos superiores de C a A
      - Mover el plato mayor de B a C
      - Mover los $n-1$ platos superiores de A a C

      Escrito formalmente como algoritmo recursivo resulta similar al del apunte, vea el algoritmo 2. Se invoca Hanoi(n, A, C).

Algoritmo 2: Solución recursiva a la variante de las torres de Hanoi

```
procedure Hanoi(n, src, dst)
    if n > 0 then
        Hanoi(n − 1, src, dst)
        Mover de src a B
        Hanoi(n − 1, dst, src)
        Mover de B a dst
        Hanoi(n − 1, src, dst)
    end
```

   b) Llamemos H_n al número de movidas en esta variante de las torres de Hanoi. Tenemos la recurrencia:

$$H_{n+1} = 2 + 3H_n \quad H_0 = 0$$

Definiendo:

$$h(z) = \sum_{n \geq 0} H_n z^n$$

obtenemos la ecuación:

$$\frac{h(z) - H_0}{z} = 3h(z) + \frac{2}{1-z}$$

Nuestro sistema de álgebra simbólica nos da h como fracciones parciales (ver el archivo `p3.mc` adjunto):

$$h(z) = \frac{1}{1-3z} - \frac{1}{1-z}$$

de donde leemos:

$$H_n = 3^n - 1$$

Puntajes

Total		35
a) Estrategia para resolver el problema	20	
b) Derivar la recurrencia	15	

4. Este es un caso clásico de aplicación de *backtracking*. Usaremos la notación estándar de grafos, en particular $N(v)$ son los vértices vecinos a v. Resulta el algoritmo 3.

Algoritmo 3: Contar número de caminos en un DAG

```
function Paths(G, s, t)
    if s = t then
        return 0
    end
    c ← 0
    for v ∈ N(s) do
        c ← c + Paths(G, v, t)
    end
    return c
```

Puntajes

Total	35
Propuesta de backtracking	10
Algoritmo detallado	25

HvB / LaTeX 2_ε