

CC30A Algoritmos y Estructuras de Datos – Control 1

Profesor: Patricio Poblete

Abril 28, 2000

1. a) Resuelva la siguiente ecuación de recurrencia:

$$a_n = 5a_{n-1} - 6a_{n-2}; \quad a_0 = 0; \quad a_1 = 1;$$

de las dos maneras siguientes:

- 1) Usando el método de la ecuación característica (esto es, suponiendo una solución de la forma λ^n).
 - 2) Usando una función generatriz.
- b) La ecuación $a_{n+1} = 2a_n + 1$, $a_0 = 0$ para el número de movidas en las Torres de Hanoi es una ecuación no homogénea. Obtenga a partir de ella una ecuación homogénea restando la ecuación para a_{n+2} menos la misma ecuación para a_{n+1} . Encuentre la condición inicial adicional que necesita, y luego resuelva la ecuación obtenida, ya sea usando el método de la ecuación característica o el de la función generatriz.
2. a) Se desea definir un tipo de listas enlazadas que permita almacenar elementos heterogéneos, por ejemplo, enteros y strings. Defina una clase abstracta `Nodo` y dos clases derivadas que permitan implementar este tipo de listas. Sus definiciones deben permitir que el siguiente código imprima correctamente todos los elementos de la lista, suponiendo que `c` apunta al primer nodo de la lista:

```
for( Nodo p=c; p != null; p=p.siguiente() )  
    System.out.println( p );
```

- b) Suponga que `c` apunta al record de encabezamiento de una lista de doble enlace. Se ha descubierto que un vándalo ha modificado todos los punteros `prev` en la lista. Escriba un trozo de programa que asigne a cada puntero `prev` su valor correcto, de modo que la lista vuelva a ser válida.
3. Normalmente, la búsqueda binaria se programa haciendo dos comparaciones por cada iteración del ciclo: una de la forma `if (x==a[k])` y otra de la forma `if (x<a[k])`. Una codificación más eficiente puede lograrse si dentro del ciclo se hace sólo *una* comparación del tipo `if (x<=a[k])`, y se realiza una comparación de igualdad sólo una vez que el ciclo ha terminado.

Programa este tipo de búsqueda binaria, explicando claramente el invariante (inicialización, preservación, terminación). Suponga que el arreglo es $a[0], \dots, a[n-1]$, que se busca un elemento x , y que el resultado debe ser el subíndice k en donde está el elemento (o $k=-1$ si no está).

Tiempo: 2 horas

Entregar en hojas separadas

Con apuntes de clases