Universidad del Norte

IST 4330 - ESTRUCTURAS DISCRETAS

Profesor: Alfonso Mancilla Herrera.

Monitor: Enrique Niebles Saco.

19 de febrero de 2021

Relaciones de recurrencia

Una relación de recurrencia es una función escrita en términos de sus predecesores, es decir:

$$f_n = f(n) = \sum_{j=1}^{k} c_j f_{n-j}$$

Donde j hace referencia a los predecesores. El grado de la función estará dado por el predecesor más lejano o distante. Así mismo, la relación de recurrencia tendrá tantos casos bases como indice el grado de la RR. Es decir, si una RR es de grado dos, esta deberá tener 2 casos base.

Nota: Los casos bases no pueden ser generador por la RR. El coeficiente $c_k \neq 0$.

Ejemplo 1 La sucesión de números de Jacobsthal está definida de la siguiente forma:

$$\langle 2, 1, 5, 7, 17, 31, 65, 127, 257, \ldots \rangle$$
$$\underbrace{f_n = 2f_{n-2} + f_{n-1};}_{RR} \quad \underbrace{f_0 = 2, f_1 = 1;}_{\text{casos bases}} \quad \underbrace{n \geq 2}_{\text{dominio}}$$

Probemos que se cumpla

```
def recurrencia(n):
  if n == 0: return 2
  elif n == 1: return 1
  return 2*recurrencia(n-2) + recurrencia(n-1)
```

```
for i in range(0, 9):
  print("n={i}. Resultado recurrente: {rr}".format(i=i, rr=recurrencia(i)))
```

```
    n=0. Resultado recurrente: 2
    n=1. Resultado recurrente: 1
    n=2. Resultado recurrente: 5
    n=3. Resultado recurrente: 7
    n=4. Resultado recurrente: 17
    n=5. Resultado recurrente: 31
    n=6. Resultado recurrente: 65
    n=7. Resultado recurrente: 127
    n=8. Resultado recurrente: 257
```

Ahora bien, las relaciones de recurrencia pueden ser de los siguientes tipos:

1. **Lineal:** El exponente de las funciones predecesoras es de grado 1. Es decir f_{n-j}^{k} donde $k = 1$
2. **Homogénea:** Todos los términos de la función deben tener un predecesor.
3. **Constante:** Ningún coeficiente puede depender de n.

El ejemplo mostrado anteriormente cumple todos estos requisitos.

Las RR son expresiones **no** cerradas. Es decir, que $f_n = f_{n-1} + f_{n-2}, \ldots$ Una expresión cerrada puede ser: $2^n, \quad 3n^3 + 2n + 1, \quad \ldots$

Lo que se busca con una relación de recurrencia es hallar su expresión **cerrada.**

Procedamos a resolver una RR. Solo si la función es **lineal, homógenea y constante** se puede usar el método de los coeficientes. De lo contrario, debe resolverse por el método de iteraciones o por funciones generadoras.

Ejemplo 2:

Dada la siguiente RR, halle su expresión no recurrente (cerrada): $f_n = f_{n-1} + 4; \quad f_0 = 3; \quad n > 1$

Dado que la RR no es homogénea, se resolverá por el método de iteraciones.

Para k = 1 (Primera iteración):

$$f_n = f_{n-1} + 4; \quad f_0 = 3; \quad n > 1$$

Para k = 2 (Segunda iteración):

$$f_n = \underbrace{(f_{(n-1)-1} + 4)}_{f_{n-1}} + 4$$

$$f_n = (f_{n-2} + 4) + 4$$

Para k = 3 (Tercera iteración):

$$f_n = \underbrace{(f_{(n-1)-2} + 4) + 4}_{f_{n-2}} + 4$$

$$f_n = (f_{n-3} + 4) + 4 + 4$$

Repites el proceso hasta encontrar un patrón. En este caso es el siguiente:

$$f_n = f_{n-k} + 4k$$

Se igualan los casos. Por lo tanto $f_{n-k} = f_0 = 3$ Lo cual implica que $n - k = 0$. Entonces $n = k$. Remplazando k

$$f_n = f_{n-n} + 4n$$

$$f_n = f_0 + 4n$$

Finalmente,

$$\boxed{f_n = 3 + 4n}$$

Ahora se comprueba que la versión no recurrente, sea igual a la recurrente.

```
def recurrencia(n):
  if n == 0: return 3
  return recurrencia(n-1) + 4

def no_recurrente(n):
  return 3 + 4*n

for i in range(0, 9):
  print("n={i}. Resultado recurrente: {rr}\tResultado no recurrente: {nrr}".format(i=i, rr=recurrencia(i), nrr=no_recurrente(i)
```

```
n=0. Resultado recurrente: 3    Resultado no recurrente: 3
```

```
n=1. Resultado recurrente: 7    Resultado no recurrente: 7
n=2. Resultado recurrente: 11   Resultado no recurrente: 11
n=3. Resultado recurrente: 15   Resultado no recurrente: 15
n=4. Resultado recurrente: 19   Resultado no recurrente: 19
n=5. Resultado recurrente: 23   Resultado no recurrente: 23
n=6. Resultado recurrente: 27   Resultado no recurrente: 27
n=7. Resultado recurrente: 31   Resultado no recurrente: 31
n=8. Resultado recurrente: 35   Resultado no recurrente: 35
```

Ejemplo 3:

Dada la relación de recurrencia:

$$f_n = f_{n-1} + 3^{n-1}; \quad n \geq 1; \quad f_1 = 2$$

Dado que la RR no es homogénea, se resolverá por el método de iteraciones.

Para k = 1 (Primera iteración):

$$f_n = f_{n-1} + 3^{n-1}; \quad n \geq 1; \quad f_1 = 2$$

Para k = 2 (Segunda iteración):

$$f_n = \underbrace{(f_{(n-1)-1} + 3^{(n-1)-1})}_{f_{n-1}} + 3^{n-1};$$

$$f_n = f_{n-2} + 3^{n-2} + 3^{n-1}$$

Para k = 3 (Tercera iteración):

$$f_n = \underbrace{(f_{(n-1)-2} + 3^{(n-1)-2} + 3^{(n-1)-1})}_{f_{n-2}} + 3^{n-1}$$

$$f_n = f_{n-3} + 3^{n-3} + 3^{n-2} + 3^{n-1}$$

Se repite hasta encontrar un patrón. En este caso:

$$f_n = f_{n-k} + \sum_{i=0}^{k-1} 3^{n-i-1}$$

Reescribiendo

$$f_n = f_{n-k} + \sum_{i=0}^{k-1} 3^n 3^{-i} 3^{-1}$$

Como 3^n y 3^{-1}, no depende de i, se pueden extraer de la sumatoria:

$$f_n = f_{n-k} + 3^{n-1} \sum_{i=0}^{k-1} 3^{-i}$$

Se identifica que estamos ante una serie geométrica. Por lo cual

$$\sum_{i=0}^{k-1} r^i = \frac{1-r^k}{1-r}$$

Continuando el ejercicio, $3^{-i} = \left(\frac{1}{3}\right)^i$

$$f_n = f_{n-k} + 3^{n-1} \sum_{i=0}^{k-1} \left(\frac{1}{3}\right)^i$$

Donde $r = \frac{1}{3}$. Entonces,

$$f_n = f_{n-k} + 3^{n-1} \frac{1-\left(\frac{1}{3}\right)^k}{1-\left(\frac{1}{3}\right)}$$

$$f_n = f_{n-k} + 3^{n-1} \frac{1-\left(\frac{1}{3}\right)^k}{\frac{2}{3}}$$

$$f_n = f_{n-k} + 3^{n-1} \frac{1-\left(\frac{1}{3}\right)^k}{2 \cdot 3^{-1}}$$

$$f_n = f_{n-k} + 3^{n} \frac{1-\left(\frac{1}{3}\right)^k}{2}$$

Se igualan los casos bases, quedando $f_{n-k} = f_1 = 2$. Por tanto $n-k=1$, entonces $k = n-1$. Reemplazando k, se obtiene que

$$f_n = f_{n-(n-1)} + 3^n \frac{1 - \left(\frac{1}{3}\right)^{n-1}}{2}$$

$$f_n = f_1 + \frac{3^n - 3^n \left(\frac{1}{3}\right)^{n-1}}{2}$$

$$f_n = 2 + \frac{3^n - 3^n \cdot (3^{-1})^{n-1}}{2}$$

Recordar que $(a^b)^c = a^{bc}$ y $a^b \cdot a^c = a^{b+c}$

$$f_n = 2 + \frac{3^n - 3^n \cdot (3)^{1-n}}{2}$$

$$f_n = 2 + \frac{3^n - 3^{n+1-n}}{2}$$

$$f_n = 2 + \frac{3^n - 3}{2}$$

$$f_n = \frac{4}{2} + \frac{3^n - 3}{2}$$

Finalmente, la versión no recurrente de la RR es:

$$\boxed{f_n = \frac{3^n + 1}{2}}$$

```
def recurrencia(n):
  if n == 1: return 2
  return recurrencia(n-1) + 3**(n-1)

def no_recurrente(n):
  return int((3**n + 1)/2)

for i in range(1, 9):
  print("n={i}. Resultado recurrente: {rr}\t Resultado no recurrente: {nrr}".format(i=i, rr=recurrencia(i), nrr=no_recurrente(i
```

```
n=1. Resultado recurrente: 2     Resultado no recurrente: 2
n=2. Resultado recurrente: 5     Resultado no recurrente: 5
```

```
n=3. Resultado recurrente: 14    Resultado no recurrente: 14
n=4. Resultado recurrente: 41    Resultado no recurrente: 41
n=5. Resultado recurrente: 122   Resultado no recurrente: 122
n=6. Resultado recurrente: 365   Resultado no recurrente: 365
n=7. Resultado recurrente: 1094  Resultado no recurrente: 1094
n=8. Resultado recurrente: 3281  Resultado no recurrente: 3281
```

Ejercicios próximas sesiones

En las siguientes dos secciones estaremos trabajando 2 ejercicios por el método de coeficientes y 3 o 4 más con el método de iteraciones.

1. $f_n = f_{n-1} + f_{n-2}; \quad f_0 = 3, f_1 = 6; \quad n \geq 2.$
2. $f_n = 2f_{n-2} + f_{n-1}; \quad f_0 = f_1 = 1; \quad n \geq 2.$
3. $f_n = 2f_{n-1} + f_{n-2} - 2f_{n-3}; \quad f_0 = 0, f_1 = 1, f_2 = 2; \quad n \geq 3.$
4. $f_n = \sqrt{1 + f_{n-1}^2}; \quad f_0 = 0; \quad n \geq 1.$
5. $f_n = 2f_{n-1} + n; \quad f_0 = 0; \quad n \geq 1.$
6. $f_n = 6f_{n-1} - 9f_{n-2}; \quad f_0 = 0, f_1 = 1; \quad n \geq 2.$
7. $f_n = \frac{3}{2}f_{n-1} + 4; \quad f_0 = 2; \quad n \geq 1.$