ESTRUCTURAS DISCRETAS

- **Professor:** Lic. Ing. Alfonso M. Mancilla Herrera, M.Sc. Dr.(c)
- **Estudiantes: {coloque aquí nombre y fotografía de los integrantes}**

Fig. 1: Fotografía del Profesor Alfonso Mancilla

- **Topics:**
  - **Relaciones de Recurrencia.**

- **Análisis Combinatorio**
- **Diseño de Funciones Recursivas**
- **Teoría de Grafos**
- **Solución de problemas de Ciencias Computacionales. (Funciones Generadoras Ordinarias y método Simbólico).**

HONESTIDAD

Observaciones

- El proyecto se desarrollará en equipos de dos o tres estudiantes.
- Los equipos pueden conformarse con un estudiante de otro grupo, previa autorización del docente.
- El nombre del .zip tiene que ser formado con los nombres de los estudiantes. Ejemplo: AlfonsoMancilla_CarlosArdila_MarleneDuarte.
- Las entregas están programadas en la siguiente forma:
  - Entrega 01 (30%). Sábado 27 de Marzo.
  - Entrega 02 (30%). Sábado 17 de Abril.
  - Entrega 03 (40%). Sábado 08 de Mayo.
- Cada entrega parcial tiene que incluir:
  1. Autoevaluación cualitativa y cuantitativa de **cada uno** de los miembros del equipo. Esta tendrá un peso del 5% y es de caracter obligatorio.
  2. Entregar sólo el archivo .zip generado por Overleaf, y aparte el .pdf correspondiente; éste debe incluir el código desarrollado en Matlab o Phyton(`\usepackage{listings}`). Recuerde incluir solamente a los integrantes del equipo que aportaron a la solución de los problemas. En la autoevaluación describa en qué medida lo hicieron.
  3. El .zip debe estar estructurado de la siguiente manera:
     - .zip con el fuente del proyecto de LaTeX.
     - .pdf del proyecto de LaTeX. Este debe estar fuera del .zip con el fuente del proyecto.
     - Una carpeta llamada **codigo** donde deben estar los programas pedidos en cada entrega.
  4. Fotografía de los miembros del equipo que trabajaron.
  5. Archivos fuente y ejecutable del código asociado con cada entrega. Nuestras alternativas serán (VBA©, Python©, Matlab©).

6. La entrega 03 tiene que incluir el archivo .zip con el informe escrito en Latex y el video (Utilice como referencia el que se muestra en este enlace: https://www.youtube.com/watch?v=ox09Jko1ErM/
7. Todas las componentes de la evaluación son de carácter obligatorio

RÚBRICA

Entregas 01 y 02. (10% cada una)

1. Autoevaluación 5%
2. Informe 20%
3. Problemas 75%

Entrega 03. (10%)

1. Autoevaluación 5%
2. Informe 20%
3. vídeo 20%
4. Problemas 55%

Rúbrica

Las soluciones a los problemas aquí propuestos deben estar detalladas.

Entrega 01

Componente		Porcentaje
Problema 1	80 %	75 %
Problema 2	20 %	
Informe	Debe estar hecho con normas IEEE, APA o EasyChair u otro.	20 %
Autoevaluación	Individual. Cada miembro del equipo debe realizar su propia atuoevaluación.	5 %

Entrega 02

Componente		Porcentaje
Problema 1	60 %	75 %
Problema 2	40 %	
Informe	Debe estar hecho con normas IEEE, APA o EasyChair u otro.	20 %
Autoevaluación	Individual. Cada miembro del equipo debe realizar su propia autoevaluación.	5 %

Entrega 01

1 Problema 01: Funciones Generadoras Ordinarias

Se recomienda consultar el capítulo 3 del libro *An Introduction to the Analysis of Algorithms* de Robert Sedgewick y Philippe Flajolet [1]. Otras fuentes bibliográficas altamente recomendables son: [2], [3], [4] y [5].

Funciones Generadoras Ordinarias

Descripción

Una función generadora es una serie de potencias cuyos coeficientes corresponden a los términos de una secuencia. De este modo, las secuencias pueden ser definidas mediante una función generadora, lo que a su vez significa que permiten transformar problemas con secuencias en problemas con funciones.
Dada una secuencia f_0, f_1, f_2, $..f_k$, ..., se define su función generadora como

$$F(z) = \sum_{k=0}^{\infty} f_k \cdot z^k = f_0 + f_1 \cdot z^1 + f_2 \cdot z^2 + \cdots + f_n \cdot z^N + \cdots \tag{1}$$

En (1) se observa que se están tomando todos los términos de la secuencia, que son infinitos, y que cada k-ésimo término está multiplicado por la k-ésima potencia de x. El término f_k representa todos los términos de la secuencia, quienes en la función generadora, pasan a ser los coeficientes. **De modo que una vez se tiene la función generadora de una secuencia, es posible hallar la fórmula para la secuencia a partir del hallazgo de los coeficientes de la función generadora**.
Por ejemplo dada la secuencia de unos, donde $f_k = 1$, es decir, todos los términos de la secuencia son 1, su función generadora será

$$F(z) = \sum_{k=0}^{\infty} 1 \cdot z^k = 1 + 1 \cdot z^1 + 1 \cdot z^2 + \cdots + 1 \cdot z^N + \cdots \tag{2}$$

Esta función generadora, corresponde a una serie geométrica donde el radio es igual a 1, y esta serie geométrica converge a un valor que corresponde a

$$F(z) = \sum_{N\geq 0} 1 \cdot z^k = \frac{1}{1-z}. \tag{3}$$

Si en lugar de una secuencia $\{1\}_{N\geq 0}$, se tuviera la secuencia de los enteros no negativos, es decir, $\{N\}_{N\geq 0}$, la función generadora sería:

$$F(z) = \sum_{N\geq 0} N \cdot z^N = 0 + 1 \cdot z^1 + 2 \cdot z^2 + \cdots + N \cdot z^N + \cdots \tag{4}$$

Al igual que la ecuación (2), esta función generadora converge a un valor

$$F(z) = \sum_{N\geq 0} N \cdot z^N = \frac{z}{(1-z)^2} \tag{5}$$

Es importante resaltar que en lo relacionado a funciones generadoras, el objetivo es hallar los coeficientes a partir de una función generadora dada, y para ello se recurre a las operaciones básicas (ver Tabla 2) relacionadas con estas y a las funciones generadoras básicas (ver Tabla 1).

En el caso de las funciones generadoras ordinarias, es importante resaltar que la manera de indicar que los coeficientes de una función generadoras es

$$[z^N]F(z) = f_N.$$

Claramente, esto significa que

$$F(z) = \sum_{N\geq 0} f_N \cdot z^n.$$

Por ejemplo:

$$[z^N]\frac{1}{1-z} = 1.$$

$$[z^N]\frac{z}{(1-z)^2} = N.$$

$1, 1, 1, \ldots, 1, \ldots$	$\dfrac{1}{1-z} = \sum_{N\geq 0} z^N$
$0, 1, 2, 3, \ldots, N, \ldots$	$\dfrac{z}{(1-z)^2} = \sum_{N\geq 0} N z^N$
$0, 0, 1, 3, 6, \ldots, \binom{N}{2}, \ldots$	$\dfrac{z^2}{(1-z)^3} = \sum_{N\geq 2} \binom{N}{2} z^N$
$0, \ldots, 0, 1, M+1, \ldots, \binom{N}{M}, \ldots$	$\dfrac{z^M}{(1-z)^{M+1}} = \sum_{N\geq M} \binom{N}{M} z^N$
$1, M+1, , \binom{M+2}{2}, \binom{M+3}{3}, \ldots, \binom{N}{M}, \ldots$	$\dfrac{1}{(1-z)^M} = \sum_{N\geq 0} \binom{N+M-1}{M} z^N$
$1, 0, 1, \ldots, 1, 0, \ldots$	$\dfrac{1}{1-z^2} = \sum_{N\geq 0} z^{2N} = \sum_{N\geq 0} \dfrac{1+(-1)^N}{2} z^N$
$2^0, 2^1, 2^2, 2^3, \ldots, 2^N, \ldots$	$\dfrac{1}{1-2z} = \sum_{N\geq 0} 2^N z^N$
$c^0, c^1, c^2, c^3, \ldots, c^N, \ldots$	$\dfrac{1}{1-cz} = \sum_{N\geq 0} c^N z^N$
$\binom{M}{0}, \binom{M}{1}, \ldots, \binom{M}{N}, \ldots, , \binom{M}{1}, \binom{M}{0}.$	$(1+z)^M = \sum_{N\geq 0} \binom{M}{N} z^N$
$0, 1, \frac{1}{2}, \frac{1}{3}, \ldots, \frac{1}{N}, \ldots$	$\ln \dfrac{1}{1-z} = \sum_{N\geq 1} \dfrac{z^N}{N}$

Tab. 1: Funciones Generadoras

En la Tabla 1 se muestran algunas funciones generadoras básicas y que mediante su manipulación, permiten hallar los coeficientes de otras funciones generadoras. Estas manipulaciones se realizan mediante el uso de las operaciones definidas para las funciones generadoras.

Suma (o resta) $F(z)+G(z)$	$\sum_{N\geq 0}(f_N+g_N)z^N$	$f_0{+}g_0,\ f_1{+}g_1,\ \ldots,\ f_n{+}g_n,\ \ldots$
Escala $F(k\cdot z)$	$\sum_{N\geq 0} f_N k^N z^N$	$f_0, kf_1, k^2f_2,\ \ldots,\ k^N f_N,\ \ldots$
Derivada $F'(z)$	$\sum_{N\geq 0}(N+1)\cdot f_{N+1}\cdot z^N$	$f_1,\ 2{\cdot}f_2,\ 3{\cdot}f_3,\ \ldots, N{\cdot}f_N\ \ldots$
Suma Parcial $\dfrac{F(z)}{1-z}$	$\sum_{N\geq 0}\left(\sum_{0\leq k\leq N} f_k\right)z^N$	$f_0,\ f_0+f_1,\ldots,\ \sum_{0\leq k\leq N} f_k,\ldots$
Convolución $F(z)\cdot G(z)$	$\sum_{N\geq 0}\left(\sum_{0\leq k\leq N} f_k\cdot g_{N-k}\right)z^N$	$f_0g_0,\ f_1g_0{+}f_0g_1,\ \ldots,\ \sum_{0\leq k\leq N} f_k\cdot g_{n-k}$
Desplazar a la derecha $zF(z)$	$\sum_{n\geq 1} f_{n-1}z^N$	$0,\ f_0,\ f_1,\ f_2,\ldots, f_n,\ldots$
Diferencia (Telescópica) $(1-z)F(z)$	$f_0+\sum_{N\geq 1}(f_N-f_{N-1})z^N$	$f_0,\ f_1{-}f_0,\ldots, f_N{-}f_{N-1},\ldots$

Tab. 2: Operaciones básicas con Funciones Generadoras

Ejemplos

El procedimiento general para hallar la función generadora $F(n)$ de una secuencia $\{f_n\}$ es el de primero definir $F(z)$, esto es:

$$F(z) = \sum_{n \geq 0} f_n \cdot z^n.$$

A partir de aquí, hay varias maneras de proceder:

- Una posible opción es la de modificar esta expresión de $F(z)$ utilizando las propiedades de las OGF para así llegar a las OGF básicas, de modo que $F(z)$ finalmente quede en función de estas.
- También es posible empezar con una OGF básica e ir aplicando distintas propiedades a estas hasta llegar a la expresión de la sumatoria de $F(z)$.

Ejemplo 1 Dada la secuencia $\left\{2^{N+1}\right\}_{N\geq 2}$, halle su función generadora.
Este ejemplo lo resolveremos usando una de las funciones generadoras básicas y aplicando propiedades sucesivamente hasta llegar a la expresión buscada.
Lo primero es definir lo siguiente:
Dada la secuencia $\left\{2^{N+1}\right\}_{N\geq 2}$, definimos $F(z)$ como su función generadora.

$$F(z) = \sum_{N \geq 2} 2^{N+1} \cdot z^N.$$

Empezaremos usando la OGF más básica:

$$\frac{1}{1-z} = \sum_{N \geq 0} z^N.$$

El procedimiento a seguir es aplicar propiedades de las OGF hasta que el lado derecho de la ecuación sea igual al buscado, es decir, la expresión de la sumatoria en $F(z)$.
Primero aplicaremos la propiedad de escala, con $k = 2$:

$$\frac{1}{1-2z} = \sum_{N \geq 0} 2^N \cdot z^N.$$

Ahora, dado que la expresión buscada tiene 2^{N+1}, multiplicamos ambos lados de la expresión por 2.

$$2 \cdot \left(\frac{1}{1-2z}\right) = 2 \cdot \left(\sum_{N \geq 0} 2^N \cdot z^N\right)$$

$$\frac{2}{1-2z} = \sum_{N \geq 0} 2^{N+1} \cdot z^N$$

La expresión de la sumatoria es muy cercana a la buscada, sin embargo, aún falta el índice de la sumatoria en la expresión que buscamos. Por lo tanto, separamos la sumatoria.
Para aclarar lo que se hará, es importante recordar lo siguiente:

$$\sum_{N\geq 0} f_N \cdot z^N = f_0 + f_1 \cdot z^1 + f_2 \cdot z^2 + f_3 \cdot z^3 + \cdots$$

y que esto puede ser reescrito de diversas maneras, y en este caso, usando la propiedad asociativa de la suma, tenemos:

$$\sum_{N\geq 0} f_N \cdot z^N = \left(f_0 + f_1 \cdot z^1\right) + \left(f_2 \cdot z^2 + f_3 \cdot z^3 + \cdots\right)$$

$$\sum_{N\geq 0} f_N \cdot z^N = \sum_{0\leq N\leq 1} f_N \cdot z^N + \sum_{N\geq 2} f_N \cdot z^N$$

nos será bastante útil. Dado que ahora reescribimos

$$\frac{2}{1-2z} = \sum_{N\geq 0} 2^{N+1} \cdot z^N$$

como

$$\frac{2}{1-2z} = \sum_{0\leq N\leq 1} 2^{N+1} \cdot z^N + \underbrace{\sum_{N\geq 2} 2^{N+1} \cdot z^N}_{F(z)}.$$

A partir de aquí, ya que tenemos $F(z)$, lo que hacemos es despejar.

$$\frac{2}{1-2z} = \sum_{0\leq N\leq 1} 2^{N+1} \cdot z^N + F(z)$$

$$F(z) = \frac{2}{1-2z} - \sum_{0\leq N\leq 1} 2^{N+1} \cdot z^N$$

$$F(z) = \frac{2}{1-2z} - 2^2 \cdot z^1 - 2^1 \cdot z^0$$

$$F(z) = \frac{2}{1-2z} - 4z - 2.$$

De aquí, tenemos que la respuesta es que la función generadora de la secuencia $\left\{2^{N+1}\right\}_{N\geq 2}$ es

$$F(z) = \frac{2}{1-2z} - 4z - 2.$$

Ejemplo 2 Dada la siguiente secuencia $\{f_N\}$, halle su OGF.

$$f_N = \begin{cases} N, & N \geq 4 \\ 1, & 1 \leq N < 4 \\ 0, & N = 0 \end{cases}$$

Primero, definimos $F(z)$ como la función generadora de la secuencia f_N.

$$F(z) = \sum_{1 \leq N \leq 3} 1 \cdot z^N + \sum_{N \geq 4} N \cdot z^N + 0 \cdot z^0$$

Este ejemplo lo resolveremos usando, nuevamente, la OGF más básica. Es importante resaltar que se puede empezar a partir de cualquier otra de las OGF, pero utilizamos esta para así mostrar más ejemplos de las operaciones.

$$\frac{1}{1-z} = \sum_{N \geq 0} z^N.$$

Primero procedemos a aplicar la operación de multiplicación por el índice a la función generadora. Por lo que al aplicar la operación siguiendo su definición en la tabla, obtenemos:

$$\frac{1}{(1-z)^2} = \sum_{N \geq 0} (N+1) \cdot z^N.$$

Aquí podemos bien o aplicar el cambio de variable $N \to N-1$ en la sumatoria o bien, aplicar la operación de desplazar a la derecha [la secuencia]. En todo caso, el resultado es el mismo:

$$\frac{z}{(1-z)^2} = \sum_{N \geq 0} N \cdot z^N \left(= \sum_{N \geq 1} N \cdot z^N \right).$$

Esta expresión ya es más cercana a la que buscamos. Sin embargo, el índice de esta sumatoria inicia en 0, y el buscado inicia en 4, por lo que procedemos a restar estos elementos:

$$\begin{aligned} \frac{z}{(1-z)^2} &= \sum_{N \geq 0} N \cdot z^N \\ &= \sum_{0 \leq N \leq 3} N \cdot z^N + \sum_{N \geq 4} N \cdot z^N \\ \frac{z}{(1-z)^2} - \sum_{0 \leq N \leq 3} N \cdot z^N &= \sum_{N \geq 4} N \cdot z^N \\ \frac{z}{(1-z)^2} - 3z^3 - 2z^2 - z &= \sum_{N \geq 4} N \cdot z^N. \end{aligned}$$

Luego de este paso, vemos que aún falta la otra sumatoria de la definición de $F(z)$ para este ejercicio. Por lo que simplemente sumamos la expresión de ambos lados y obtenemos:

$$\frac{z}{(1-z)^2} - 3z^3 - 2z^2 - z + \sum_{1\leq N\leq 3} 1\cdot z^N = \underbrace{\sum_{N\geq 4} N\cdot z^N + \sum_{1\leq N\leq 3} 1\cdot z^N}_{F(z)},$$

y aquí una vez más ya tenemos nuestro $F(z)$.

$$\begin{aligned} F(z) &= \frac{z}{(1-z)^2} - 3z^3 - 2z^2 - z + \sum_{1\leq N\leq 3} 1\cdot z^N \\ &= \frac{z}{(1-z)^2} - 3z^3 - 2z^2 - z + z + z^2 + z^3 \\ F(z) &= \frac{z}{(1-z)^2} - 2z^3 - z^2. \end{aligned}$$

Por lo que finalmente la solución a este ejemplo es que la OGF de la secuencia

$$f_N = \begin{cases} N, & N \geq 4 \\ 1, & 1 \leq N < 4 \\ 0, & N = 0 \end{cases}$$

es

$$F(z) = \frac{z}{(1-z)^2} - 2z^3 - z^2.$$

1.1 Funciones Generadoras y Secuencias.

Dadas las siguientes secuencias, hallar la función generadora de cada una.

1. [*] Dada la secuencia $\{4 \cdot n^3\}_{n\geq 2}$, halle su función generadora ordinaria.

2. [*] Dada la secuencia $\{n \cdot 2^{n+3}\}_{n\geq 3}$, halle su función generadora ordinaria.

3. [*] Dada la secuencia
$$a_n = \begin{cases} n, & n \bmod 2 = 0 \\ -n, & n \bmod 2 = 1 \end{cases},$$
halle su función generadora ordinaria.

4. [**] Dada la secuencia
$$\left\{ \sum_{0\leq k\leq n} k \cdot (-1)^n \cdot (-1)^k \right\}_{n\geq 3},$$
halle su función generadora ordinaria.

5. [**] Dada la secuencia
$$\begin{cases} 0, & n \leq k \\ \sum_{0\leq i\leq n} (i+1), & n > k \end{cases},$$
halle su función generadora ordinaria para un k dado. $k \in \mathbb{N}$.

6. [*] Dada la secuencia
$$\begin{cases} n^2 \cdot 2^{n+1}, & n \neq 5 \\ 3^n, & n = 5 \end{cases},$$
halle su función generadora ordinaria.

7. [*] Dada la siguiente función generadora ordinaria, halle la respectiva secuencia.
$$A(z) = \frac{1}{(1+2z)^5}$$

8. [***] Dada la siguiente función generadora ordinaria, halle la respectiva secuencia.
$$A(z) = (1-z)^2 \cdot \ln \frac{1}{1-z}$$

1.2 Python I

Para cada uno de los siguientes items, desarrolle una función con el nombre y argumentos indicados. Cada función debe devolver una **tupla** donde el primer elemento sea el total de elementos que cumplen con la condición indicada; el segundo elemento de la tupla debe ser una **lista** con los elementos que cumplen la condición.

1. - Nombre de la función: `sin_repetidos`
   - Argumentos: `a, b`
   - Condición: Números enteros positivos entre a y b, inclusive, que no tienen dígitos repetidos.
2. - Nombre de la función: `sin_ceros`
   - Argumentos: `a, b`
   - Condición: Números enteros positivos entre a y b, inclusive, que no tienen al 0 como dígito.
3. - Nombre de la función: `ascendentes`
   - Argumentos: `a, b`
   - Condición: Números enteros positivos entre a y b, inclusive, que tienen sus dígitos de manera **estrictamente ascendente**. Por ejemplo: 1468 cumple la condición, pero 14668 no la cumple.
4. - Nombre de la función: `conjuntos_sin_consecutivos`
   - Argumentos: `a, b, k`
   - Condición: Subconjuntos de k elementos escogidos sin repetición desde a hasta b tal que no tengan elementos consecutivos. Por ejemplo, dado $a = 1$, $b = 5$ y $k = 2$, conjuntos posibles son: $\{\{1,3\},\{1,4\},\{1,5\},\{2,4\},\{2,5\},\{3,5\}\}$. En su implementación, esto debe ser una lista con cada subconjunto, y a su vez, cada subconjunto es una lista.

Nota: Para esta entrega no pueden hacer uso de ninguna librería de Python (itertools, etc). El objetivo es que se familiaricen bastante bien con las listas en Python y otros elementos básicos del lenguaje.

Entregable

Cada equipo debe enviar un **único** archivo en formato *.zip* que por dentro tenga:

- PDF de su informe. Recuerden que el informe debe incluir la solución de cada unos de los ejercicios planteados, también debe incluir la explicación de su código en Python y pruebas de este mismo (un ejemplo por cada item). También debe incluir las autoevaluaciones cuantitativas y cualitativas de cada uno de los miembros del equipo.

- Archivo **.zip** con el fuente de su informe hecho en LaTeX.
- Una carpeta llamada **codigo** donde incluyan sus archivos de Python. Deben tener un archivo `main.py` donde estén las funciones indicadas en el ejercicio. Pueden crear otras funciones y otros archivos para apoyarse, pero es importante que en el archivo `main.py` estén las 4 funciones pedidas.

2 Entrega 02: Análisis Combinatorio

2.1 Conteo usando funciones generadoras ordinarias

Las funciones generadoras ordinarias (OGF) nos permiten resolver problemas relacionados con conteo, de una manera sencilla; de modo que haciendo manipulaciones sencillas sobre estas, problemas que sería difícil resolver directamente, son resueltos por medio de operaciones mucho más simples como la suma de polinomios o la multiplicación de estos.
Recordemos que la función generadora $F(z)$ de una secuencia f_n es de la forma

$$F(z) = \sum_{n\geq 0} f_n \cdot z^n$$

Es importante tener en cuenta que de aquí en adelante f_n siempre hace referencia al número de conjuntos de n objetos idénticos.
En esta sección se mostrará que este f_n puede ser asociado a un problema de conteo y que por medio de el uso de OGF, se puede hallar una expresión cerrada para f_n fácilmente. En estos problemas, obtendremos la OGF de f_n y luego hallaremos el coeficiente de esta función generadora. Es decir, a a partir de $F(z)$ hallaremos f_n. Recordemos la notación $[z^n]F(z) = f_n$.
En este tipo de ejercicios, las siguientes funciones generadoras suelen ser bastante comunes:

$$[z^n]\frac{1}{(1-z)^m} = \binom{m+n-1}{n}$$
$$[z^n]\frac{z^m}{(1-z)^{m+1}} = \binom{n}{m}$$

Veamos algunos ejemplos.

Ejemplo 1 Sea f_n el número de maneras de **seleccionar** n pelotas a partir de infinitas pelotas verdes, azules y doradas.
Es posible plantear el problema como una ecuación entera, de tal modo que f_n sea el número de soluciones a esta ecuación. En este caso, es:

$$S_{\text{v}} + S_{\text{a}} + S_{\text{d}} = n, \quad 0 \leq S_{\text{v}},\, S_{\text{a}},\, S_{\text{d}}.$$

Donde S_{v} representa el número de pelotas verdes, S_{a} representa el número de pelotas azules y S_{d} representa el número de pelotas doradas. Claramente, no puede haber un número de pelotas negativo. La ecuación dicta que la suma de la cantidad de pelotas debe ser igual a n, en este caso, sin restricciones, tal como indica la definición de f_n.
Este problema puede plantearse de la siguiente manera, sin recurrir a funciones generadoras:

$$f_n = \sum_{S_{\text{v}}+S_{\text{a}}+S_{\text{d}}=n} 1,$$

sin embargo, al aproximarnos a la solución utilizando funciones generadoras, el proceso será más sencillo que si se tratara de empezar utilizando esta expresión.

2.1.1 Solución usando OGF

La solución a este tipo de problemas usando OGF consiste en los siguientes pasos:

1. El primero paso a hacer es el de **modelar el problema como una ecuación entera, incluyendo las restricciones**.
2. El segundo paso a realizar es el de plantear la función generadora $F(z)$ para f_n a partir de la ecuación planteada.
3. Y finalmente, se procede a hallar el coeficiente de $F(z)$ para así hallar una expresión para f_n.

Sea $f_n \equiv$ el número de maneras de **seleccionar** n pelotas a partir de infinitas pelotas verdes, azules y doradas, entonces la ecuación que modela este problema es

$$S_{\mathrm{v}} + S_{\mathrm{a}} + S_{\mathrm{d}} = n, \quad 0 \leq S_{\mathrm{v}},\, S_{\mathrm{a}},\, S_{\mathrm{d}}.$$

Una vez se ha modelado el problema como una ecuación entera, definimos $F(z)$ como la función generadora de f_n y la expresamos a partir de la ecuación planteada. Esto es:

$$F(z) = \left(\sum_{S_{\mathrm{v}}\geq 0} z^{S_{\mathrm{v}}}\right) \cdot \left(\sum_{S_{\mathrm{a}}\geq 0} z^{S_{\mathrm{a}}}\right) \cdot \left(\sum_{S_{\mathrm{d}}\geq 0} z^{S_{\mathrm{d}}}\right).$$

En este caso usamos las mismas variables usadas en la ecuación, pero también es posible usar la letra n:

$$F(z) = \underbrace{\left(\sum_{n\geq 0} z^{n}\right)}_{S_{\mathrm{v}}} \cdot \underbrace{\left(\sum_{n\geq 0} z^{n}\right)}_{S_{\mathrm{a}}} \cdot \underbrace{\left(\sum_{n\geq 0} z^{n}\right)}_{S_{\mathrm{d}}}.$$

La elección es cuestión de preferencia. Pero es importante, en cualquiera de los dos casos, señalar a qué variable de la ecuación corresponde cada sumatoria.
Una vez planteado este $F(z)$, pasamos a hacer uso de las OGF básicas. En este caso, sabemos que $\sum_{n\geq 0} z^n = \dfrac{1}{1-z}$, por lo tanto:

$$\begin{aligned} F(z) &= \left(\sum_{n\geq 0} z^{n}\right) \cdot \left(\sum_{n\geq 0} z^{n}\right) \cdot \left(\sum_{n\geq 0} z^{n}\right) \\ &= \left(\sum_{n\geq 0} z^{n}\right)^3 = \left(\frac{1}{1-z}\right)^3. \end{aligned}$$

Aquí, hemos llegado a que
$$F(z) = \frac{1}{(1-z)^3}.$$
Una vez llegamos a una expresión de este tipo para $F(z)$, el siguiente paso es hallar el coeficiente f_n de esta expresión. Aquí, nuevamente recurrimos a la tabla de OGF básicas. Y recordemos que
$$[z^n]\frac{1}{(1-z)^m} = \binom{m+n-1}{n},$$
es decir, el coeficiente de OGF de esta forma es $\binom{m+n-1}{n}$. Aplicando esto a este ejemplo, con $m = 3$, tenemos que: $[z^n]F(z) = \binom{3+n-1}{n}$, es decir:
$$f_n = \binom{3+n-1}{n}.$$
Con lo que finalmente hemos resuelto este ejemplo.

Ejemplo 2 Sea f_n el número de maneras de distribuir n pelotas en 3 cajas, de tal modo que en las dos primeras cajas haya al menos una pelota, halle una expresión para f_n.
Solución: La ecuación que modela este problema es la siguiente:
$$S_{\text{caja}\,1} + S_{\text{caja}\,2} + S_{\text{caja}\,3} = n, \quad S_{\text{caja}\,1} \geq 1,\ S_{\text{caja}\,2} \geq 1,\ S_{\text{caja}\,3} \geq 0.$$
Ya que tenemos la ecuación, procedemos a plantear la función generadora a partir de esta ecuación.
$$F(z) = \underbrace{\left(\sum_{n\geq 1} z^n\right)}_{S_{\text{caja}\,1}} \cdot \underbrace{\left(\sum_{n\geq 1} z^n\right)}_{S_{\text{caja}\,2}} \cdot \underbrace{\left(\sum_{n\geq 0} z^n\right)}_{S_{\text{caja}\,3}}$$
aquí reescribimos las expresiones hasta llegar a una función conocida.
$$\begin{aligned}
F(z) &= \left(\sum_{n\geq 0} z^n - z^0\right) \cdot \left(\sum_{n\geq 0} z^n - z^0\right) \cdot \left(\sum_{n\geq 0} z^n\right) \\
&= \left(\frac{1}{1-z} - 1\right) \cdot \left(\frac{1}{1-z} - 1\right) \cdot \left(\frac{1}{1-z}\right) \\
&= \left(\frac{1}{1-z} - 1\right)^2 \cdot \left(\frac{1}{1-z}\right) = \left(\frac{1}{(1-z)^2} - \frac{2}{1-z} + 1\right) \cdot \left(\frac{1}{1-z}\right) \\
F(z) &= \frac{1}{(1-z)^3} - \frac{2}{(1-z)^2} + \frac{1}{1-z}
\end{aligned}$$

En este punto, nuevamente usamos la OGF

$$[z^n]\frac{1}{(1-z)^m} = \binom{m+n-1}{n},$$

y la aplicamos para obtener:

$$\begin{aligned} F(z) &= \underbrace{\frac{1}{(1-z)^3}}_{m=3} - \underbrace{\frac{2}{(1-z)^2}}_{m=2} + \underbrace{\frac{1}{1-z}}_{m=1} \\ [z^n]F(z) &= \binom{3+n-1}{n} - 2\cdot\binom{2+n-1}{n} + \binom{1+n-1}{n}. \end{aligned}$$

Por lo que finalmente, la solución a este ejercicio es:

$$f_n = \binom{3+n-1}{n} - 2\cdot\binom{2+n-1}{n} + \binom{1+n-1}{n}.$$

Ejemplo 3

¿De cuántas maneras se pueden distribuir n objetos idénticos en 2 cajas, de tal modo que en la primera caja haya un número par de objetos?

Solución La ecuación que modela este problema es:

$$S_1 + S_2 = n, \quad S_1 \bmod 2 = 0,\ S_2 \geq 0.$$

A partir de esta ecuación, definimos $F(z)$ como la función generadora de f_n y viene dada por

$$F(z) = \underbrace{\left(\sum_{\substack{n\geq 0 \\ n \bmod 2 = 0}} z^n\right)}_{S_1} \cdot \underbrace{\left(\sum_{n\geq 0} z^n\right)}_{S_2},$$

utilizando las OGF conocidas, tenemos:

$$\begin{aligned} F(z) &= \left(\sum_{n\geq 0} z^{2n}\right)\cdot\left(\sum_{n\geq 0} z^n\right) \\ &= \left(\frac{1}{1-z^2}\right)\cdot\left(\frac{1}{1-z}\right) = \frac{1}{(1-z^2)\cdot(1-z)} = \frac{1}{(1-z)\cdot(1+z)\cdot(1-z)} \\ &= \frac{1}{(1-z)^2\cdot(1+z)}. \end{aligned}$$

En este punto, aplicamos fracciones parciales y obtenemos algo de la forma

$$\begin{aligned} F(z) &= \frac{Az+B}{(1-z)^2} + \frac{C}{1+z} \\ &= \frac{Az}{(1-z)^2} + \frac{B}{(1-z)^2} + \frac{C}{1+z} \end{aligned}$$

y aquí, empleando las OGF básicas, se obtiene:

$$f_n = A \cdot \binom{n}{1} + B \cdot \binom{2+n-1}{n} + C \cdot (-1)^n.$$

Aquí reemplazamos los coeficientes de las fracciones parciales, para mayor facilidad:

$$A = -\frac{1}{4},\ B = \frac{3}{4},\ C = \frac{1}{4}.$$

Entonces

$$f_n = -\frac{\binom{n}{1}}{4} + \frac{3 \cdot \binom{2+n-1}{n}}{4} + \frac{(-1)^n}{4}$$
$$f_n = \frac{3 \cdot \binom{2+n-1}{n} - \binom{n}{1} - (-1)^n}{4}.$$

Método 2: También es posible seguir de la siguiente manera, usando la operación de suma parcial y aplicando algunas sumatorias conocidas.

$$F(z) = \left(\sum_{n\geq 0} z^{2n}\right) \cdot \left(\sum_{n\geq 0} z^n\right) = \frac{1}{1-z} \cdot \left(\frac{1}{1-z^2}\right)$$

aquí es necesario definir el coeficiente de $\frac{1}{1-z^2}$, que sabemos se puede escribir como

$$[z^n]\frac{1}{1-z^2} = \begin{cases} 1, & n \bmod 2 = 0 \\ 0, & n \bmod 2 = 1 \end{cases},$$

es decir, es 1 para los valores pares de n y 0 para los valores impares de n. Sin embargo, una expresión cerrada y que es más útil en este ejercicio, es la siguiente:

$$[z^n]\frac{1}{1-z^2} = \frac{1+(-1)^n}{2}.$$

Entonces, aplicando esto sobre $F(z)$ obtenemos:

$$F(z) = \frac{1}{1-z} \cdot \left(\sum_{n\geq 0} \frac{1+(-1)^n}{2} \cdot z^n\right).$$

Recordemos que la suma parcial está definida así:

$$\frac{1}{1-z}\cdot\sum_{n\geq 0} g_n\cdot z^n = \sum_{n\geq 0}\sum_{0\leq k\leq n} g_k\cdot z^n.$$

Entonces, en este ejemplo, $g_n = \dfrac{1+(-1)^n}{2}$, entonces $g_k = \dfrac{1+(-1)^k}{2}$ y por ende

$$F(z) = \sum_{n\geq 0}\underbrace{\sum_{0\leq k\leq n}\frac{1+(-1)^k}{2}}_{f_n}\cdot z^n,$$

por lo tanto,

$$\begin{aligned}
f_n &= \sum_{0\leq k\leq n}\frac{1+(-1)^k}{2} = \frac{1}{2}\cdot\left(\sum_{0\leq k\leq n} 1 + \sum_{0\leq k\leq n}(-1)^k\right) = \frac{1}{2}\cdot\left((n+1) + \frac{1-(-1)^{n+1}}{1-(-1)}\right)\\
&= \frac{1}{2}\cdot\left((n+1) + \frac{1-(-1)^{n+1}}{2}\right) = \frac{2n+2+1-(-1)^{n+1}}{4}\\
f_n &= \frac{2n+3-(-1)^{n+1}}{4}.
\end{aligned}$$

Con lo que finalmente, obtenemos la respuesta a este ejercicio:

$$f_n = \frac{2n+3-(-1)^{n+1}}{4}.$$

Es fácilmente demostrable que esta expresión es igual de válida que la obtenida con el primer método.

2.2 Ejercicios

1. ¿De cuántas maneras se pueden repartir $n \geq 10$ objetos idénticos en cinco cajas de tal modo que haya al menos 2 objetos en cada caja?

2. ¿De cuántas maneras se pueden distribuir n caramelos a 3 niños si cada uno obtiene una cantidad par de caramelos?

3. ¿De cuántas maneras puede la suma de 5 dados dar $5 \leq n \leq 30$ como resultado?

   `Protocolo`

   $$\sum_{i=1}^{5} x_i = n; \quad 1 \leq x_i \leq 6; \quad 5 \leq n \leq 30.$$

   `Estrategia: Directa con Palomar.`

   `Prototipo`

   $\{\ \underline{1}\ \ \underline{1}\ \ \underline{1}\ \ \underline{1}\ \ \underline{1}\ \} - \{\ \underline{1+6}\ \ \underline{1}\ \ \underline{1}\ \ \underline{1}\ \ \underline{1}\ \}$

   $$n(S) = \left[\binom{5}{5}\cdot\binom{5}{5}\right]\cdot\left[\binom{n-5}{n-5}\cdot\binom{5+(n-5)-1}{n-5}\right] -$$

   $$\left[\binom{5}{1}\cdot\binom{7}{7}\right]\cdot\left[\binom{4}{4}\cdot\binom{4}{4}\right]\cdot\left[\binom{n-7-4}{n-7-4}\cdot\binom{5+(n-11)-1}{n-11}\right]$$

   $$n(S) = \binom{n-1}{n-5} - \left[\binom{5}{1}\cdot\binom{n-7}{n-11}\right]; \quad n >= 11.$$

4. ¿De cuántas maneras se pueden distribuir n objetos idénticos en 3 cajas distintas, con al menos un objeto en las 2 primeras cajas y una cantidad impar en la tercera caja?

5. ¿De cuántas maneras se pueden distribuir n objetos idénticos en 3 cajas distintas, con al menos dos objetos en las 2 primeras cajas y una cantidad par en la tercera caja?

6. ¿De cuántas maneras se pueden recoger n pesos de 2 niños y 2 adultos si cada niño da al menos 1 peso y cada adulto al menos 3 pesos?

7. Sea f_n el número de maneras de distribuir n pelotas en 2 cajas, de tal modo que en la primeras cajas haya una cantidad par de pelotas y en la segunda caja una cantidad impar de pelotas, halle una expresión para f_n.

8. Sea f_n el número de maneras de distribuir n pelotas en 2 cajas, de tal modo que en la primeras cajas haya una cantidad par de pelotas y en la segunda caja una cantidad impar de pelotas, pero en ambas cajas al menos 2 pelotas, halle una expresión para f_n.

2.3 Problema 2: Lenguajes naturales y cadenas de Markov

Implementar un modelo de Markov de orden K a partir de un archivo de texto. A partir de este modelo, deberán generar texto pseudo-aleatorio.

Modelos de Markov Un modelo de Markov de orden $K = 0$ asume que cada caracter tiene una probabilidad fija, independiente de los caracteres anteriores. Por ejemplo, si el texto base es ”ababbabcdd”, entonces las probabilidades asignadas son:

$$Pr(X = a) = \frac{3}{10} = 0.30, \qquad Pr(X = b) = \frac{4}{10} = 0.40,$$
$$Pr(X = c) = \frac{1}{10} = 0.10, \qquad Pr(X = d) = \frac{2}{10} = 0.20.$$

Es decir, a la hora de generar texto, el siguiente caracter será generado a partir de las probabilidades ya calculadas. Este ejemplo es válido para un modelo de Markov de orden $K = 0$. Sin embargo, para diferentes valores de K, lo que sucede es que las probabilidades dependen de los K caracteres anteriores. Por ejemplo, si la cadena es ”abbbcabcbbbcb” y $K = 2$, entonces habrá cinco K-cadenas distintas: ab, ca, cb, bc y bb. Y a partir de estas cadenas, se determinan las probabilidades. Esto es: dada una K-cadena, cuál es la probabilidad de que el siguiente caracter sea X. En este ejemplo, las probabilidades serían:

Llave	Probabilidades
ab	c: $1/2 = 0.5$; b: $1/2 = 0.5$
ca	b: $1/1 = 1.0$
cb	b: $1/1 = 1.0$
bc	a: $1/3 \approx 0.33$; b: $2/3 \approx 0.67$
bb	c: $2/4 = 1/2 = 0.5$; b: $2/4 = 1/2 = 0.5$

Tab. 3: Tabla de probabilidades

Esto se interpreta de este modo: Si en el texto actual, los últimos 2 caracteres son ”bc”, el siguiente caracter será ”a” con un 33.3 % de probabilidad, o ”b” con un 66.7 % de probabiilidad.
A partir de esto, deben realizar un programa en Python que haga lo siguiente:

- Leer un archivo *.txt* de entrada y los parámetro K y N. A partir del texto y del parámetro K, construir un modelo de Markov de orden K. Y usando este modelo, genere un texto de N caracteres, donde los primeros K caracteres del texto generado son los mismos K primeros caracteres del archivo de texto.
- Debe permitir visualizar un histograma de las frecuencias de las K-tuplas de caracteres.

El programa debe permitir seleccionar si se desea visualizar el histograma o si se desea generar un texto, pero permitiendo cambiar de opción sin tener que re-abrir el programa.
En este código de Python sí está permitido utilizar librerías como NumPy. Y **deben** explicar en su informe de qué manera usaron estas librerías y qué tan útiles fueron.

Entregable

Cada equipo debe enviar un **único** archivo en formato *.zip* que por dentro tenga:

- PDF de su informe. Recuerden que el informe debe incluir la solución de cada unos de los ejercicios planteados, también debe incluir la explicación de su código en Python y pruebas de este mismo. También debe incluir las autoevaluaciones cuantitativas y cualitativas de cada uno de los miembros del equipo.

- Archivo **.zip** con el fuente de su informe hecho en LaTeX.

- Una carpeta llamada **codigo** donde incluyan sus archivos de Python. Deben tener un archivo, que debe ser el principal, llamado `main.py`, pero no se descarta que tengan archivos de Python adicionales.

3 Entrega 03: Bot de Telegram. Diseño de funciones recursivas.

3.1 Bot de Telegram

Para los siguientes problemas deben crear **UN** bot de Telegram (ver [6])
El bot debe implementar las funcionalidades descritas a continuación. Es **obligatorio** un comando `\ayuda` que describa qué hace el bot y qué comandos tiene. Además, en cada punto debe haber validación de los datos ingresados por el usuario. Cada comando debe mostrar o facilitar la comprensión del formato o la forma en que el usuario ingresa los datos.

- Recibir los coeficientes de un polinomio característico de una relación de recurrencia y mostrar cuál sería la forma de la solución. Por ejemplo, el bot debe responder al usuario algo de la forma $f(n) = c_0 \cdot 2^n + c_1 \cdot n \cdot 3^n + c_2 \cdot 3^n$.
- [**Bono +0.2**] Reciba también los casos base de la recurrencia y muestre la expresión con los valores de las constantes (c_0, c_1, $\ldots$,).
- El usuario digita una secuencia de números enteros mayores o iguales a 0, ordenados de menor a mayor. El bot debe responder al usuario con una subsecuencia de la ingresada por el usuario de tal modo que la subsecuencia pueda considerarse como una secuencia de Fibonacci. Por ejemplo: El usuario digita `2 3 4 5 7 11 13 18 22 29` y el bot responde con la subsecuencia `3 4 7 11 18 29` o también podría responder con la subsecuencia `2 3 5`.
- La entrada del usuario consiste en 3 números: El número de vértices (V), número de aristas (E) y máximos de aristas por vértice (K). En este punto deben crear un grafo simple no dirigido con V vértices, y E aristas asignadas aleatoriamente, de tal modo que ningún vértice tenga un grado mayor a K. El bot debe retornar una imagen del grafo.

Diseño de funciones recursivas 2D

Diseñe e implemente arreglos recursivos 2D, utilizando la metodología empleada en clase

1. Un if para el caso base.
2. Un elif por cada singularidad.
3. Un else para los llamados recursivos más pequeños, internamente se repite el proceso anterior, dependiendo del número.

para los siguientes casos:

1. Triángulo de Pascal, Rotado 90-180-270 grados. Por capas, ramas y diagonales.
2. Rombo por capas y ramas alternadas
3. Cubo por capas. Los tres casos
4. Home
5. Rotar Big Z

LA IMPLEMENTACIÓN TIENE QUE SER UNA COPIA EXACTA DEL DISEÑO

3.2 Video

En esta entrega deben realizar un video explicando su solución a los problemas antes planteados para esta entrega, además, en el video deben realizar una demostración del funcionamiento de su bot. El video **no** debe sobrepasar los 20 minutos.

Entregable

- Versión en PDF del informe. El informe debe tener:
  - Explicación de su código.
  - Autoevaluación cuantitivia y cualitativa de cada uno de los integrantes.
- Código fuente en LaTeXdel informe, comprimido en formato `.zip`.
- Los archivos de Python correspondientes al bot. Adicionalmente, **deben** incluir el archivo *requirements.txt* que lista los paquetes necesarios para la correcta ejecución de sus archivos de Python. Se recomienda ver **este enlace** para consultar sobre el formato de este archivo; pues los paquetes requeridos se instalarán usando el comando `pip install -r requirements.txt`.
- Todo esto debe estar contenido en un **único** archivo `.zip` que enviarán por la plataforma.

Ayuda

- $a^m \cdot a^n = a^{m+n}$, $\dfrac{a^n}{b^n} = a \cdot b^{-n}$.
- Sobre la notación para las sumatorias

$$\sum_{k=0}^{n} a_k = \sum_{0 \le k \le n} a_k = a_0 + a_1 + a_2 + \cdots + a_n$$

- Un ejemplo de transformaciones que se pueden aplicar a sumatorias

$$\sum_{0 \le k \le n} a_k = \sum_{0 \le k-\Delta \le n} a_{k-\Delta} = \sum_{\Delta \le k \le n+\Delta} a_{k-\Delta}.$$

En el siguiente ejemplo $a_k = (k-1) \cdot 2^{k-1}$ y $\Delta = -1$

$$\sum_{1 \le k \le n} (k-1) \cdot 2^{k-1} = \sum_{1 \le k+1 \le n} (k+1-1) \cdot 2^{k+1-1} = \sum_{0 \le k \le n-1} k \cdot 2^k$$

- Útil para cancelar impares. Con un pequeño cambio también funciona para cancelar pares.

$$\frac{1+(-1)^n}{2} = \begin{cases} 1, & n \bmod 2 = 0, \\ 0, & n \bmod 2 = 1 \end{cases}$$

- Sea $A(z)$ la OGF de la secuencia $\{a_n\}$, entonces

$$\frac{A^{(n)}(z)}{n!} = a_n,$$

es decir, la n-ésima derivada de la función generadora dividida entre $n!$ corresponde al n-ésimo término de la secuencia (a_n).

- Suma de una progresión aritmética

$$\sum_{0 \le k \le n} k = \frac{n \cdot (n+1)}{2} = \binom{n+1}{2}$$

- Suma de una progresión geométrica

$$\sum_{0 \le k \le n} x^k = \frac{1-x^{n+1}}{1-x}$$

References

[1] R. Sedgewick and P. Flajolet, *An introduction to the analysis of algorithms*, 2nd ed. Addison-Wesley, OCLC: ocn821694654.

[2] E. Lehman, F. Leighton, and A. Meyer, "Generating functions," in *Mathematics for Computer Science*, 2010, pp. 355–378. [Online]. Available: https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2010/readings/MIT6_042JF10_chap12.pdf

[3] D. E. Knuth, "The art of computer programming. volume 1. fundamental algorithms. third edition," p. 681.

[4] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete mathematics: a foundation for computer science*, 2nd ed. Addison-Wesley.

[5] H. S. Wilf, "Chapter 1 - introductory ideas and examples," in *Generatingfunctionology*, H. S. Wilf, Ed. Academic Press, pp. 1–26. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780127519555500046

[6] Telegram bot API. Library Catalog: core.telegram.org. [Online]. Available: https://core.telegram.org/bots/api