PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2133 Estructuras de datos y algoritmos
1° **semestre 2017**

Ayudantía I1

1. Backtracking: ZUMA

Tenemos dos listas de caracteres α y β y queremos eliminar todos los caracteres de α. // La única operación que podemos hacer para lograr nuestro objetivo es insertar el primer elemento de β en cualquier posición de α. Esto puede provocar los siguientes efectos:

- Si el caracter insertado forma un grupo de 3 o más caracteres, el grupo de elimina y se unen los extremos
- Si al unir dos extremos, los caracteres que se unen son del mismo tipo, entonces también se eliminan y se unen los extremos otra vez.

Diga si los siguientes ejemplos corresponden a una poda o una heurística y analice el costo de implementarlos y su correctitud.

1. Si el caracter a insertar no se encuentra en α, se pone al final.
2. Si el caracter a insertar no está ni en α ni en β, el estado es irresoluble.
3. Si hay solo una copia del elemento a insertar en α y no quedan más en β, el estado es irresoluble.
4. Al probar insertar en distintas posiciones, si inserto el caracter junto a otro caracter igual, solo pruebo al lado izquierdo.
5. Al modelar el problema, represento los caracteres de alpha como $(char, number)$ para no probar dentro de secuencias muy grandes.

2. Programación dinámica

Como todos saben (o deberían saber), el cálculo de $\binom{n}{k}$ se obtiene de $\frac{n!}{(n-k)!\cdot k!}$
Sin embargo, hay otra forma de calcular este valor, la cual se puede lograr con el triángulo de pascal:

$$\binom{0}{0}$$

$$\binom{1}{0} \quad \binom{1}{1}$$

$$\binom{2}{0} \quad \binom{2}{1} \quad \binom{2}{2}$$

$$\binom{3}{0} \quad \binom{3}{1} \quad \binom{3}{2} \quad \binom{3}{3}$$

$$\binom{4}{0} \quad \binom{4}{1} \quad \binom{4}{2} \quad \binom{4}{3} \quad \binom{4}{4}$$

$$\binom{5}{0} \quad \binom{5}{1} \quad \binom{5}{2} \quad \binom{5}{3} \quad \binom{5}{4} \quad \binom{5}{5}$$

$$\binom{6}{0} \quad \binom{6}{1} \quad \binom{6}{2} \quad \binom{6}{3} \quad \binom{6}{4} \quad \binom{6}{5} \quad \binom{6}{6}$$

Usando este triángulo, se puede calcular de la siguiente manera:

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

- Proponga una manera recursiva de calcular el coeficiente binomial.
- Mejore su forma recursiva para que tome tiempo polinomial.
- ¿Por qué podría servir calcular el coeficiente binomial de esta manera y no con los factoriales?

3. Ordenación

1. Con respecto a HeapSort:
   - ¿Qué ventajas y desventajas tiene?
   - Encuentre un ejemplo de datos a ordenar con heapsort que no sea estable.
   - ¿Hay alguna manera de convertir heapsort en un algoritmo estable? ¿Se puede hacer sin perder sus propiedades?
   - En un heap ¿Las operaciones de insert y pop son permutables? (Si inserto un elemento a y luego saco un elemento b ¿El heap queda igual si primero saco b y luego inserto a?)
2. Digamos que existe un algoritmo que permite hacer merge entre dos listas ordenadas en $O(1)$. Calcule la complejidad de MergeSort usando ese algoritmo.
3. Digamos que existe una estructura de datos que permite insertar un dato de manera ordenada en O(1). Calcule la complejidad de insertion sort usando esta estructura para mantener los datos que ya fueron ordenados. Explique por qué no se puede insertar en O(1) en una lista ligada ni en un array.