.UBAfiuba
FACULTAD DE INGENIERÍA

Programación dinámica: cambio mínimo

Teoría de Algoritmos I (75.29 / 95.06)
Ing. Víctor Daniel Podberezski
vpodberezski@fi.uba.ar

Cambio mínimo (revisado y revisitado)

Contamos con

Un conjunto de monedas de diferente denominación sin restricción de cantidad

$\$ = (c_1, c_2, c_3, \ldots, C_n)$

Un importe X de cambio a dar

Queremos

Entregar la menor cantidad posible de monedas como cambio

Solución greedy

No existe

una solución optima greedy general

Casos puntuales con valores “canónicos” funcionan

Solución por fuerza bruta

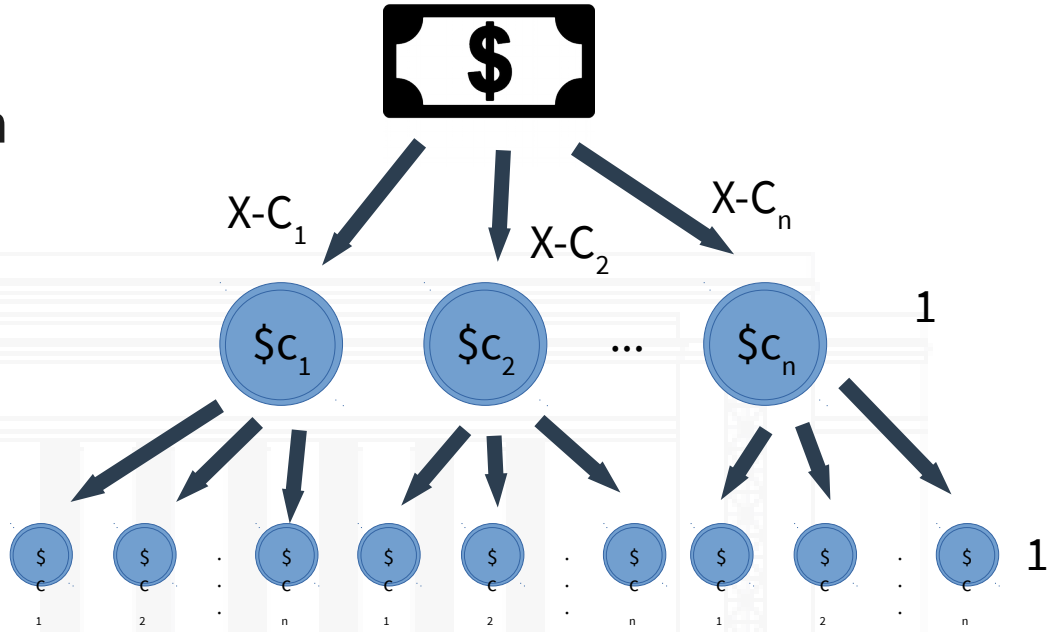
Podemos realizar un árbol de decisión

Iniciamos la raíz en el cambio X a dar

Por cada denominación posible

Dar 1 moneda de C_i

Generar sub-problema de decisión $X\text{-}C_i$



El camino a la hoja con menor profundidad

Es la menor cantidad de monedas a dar

Complejidad

$O(X^n)$

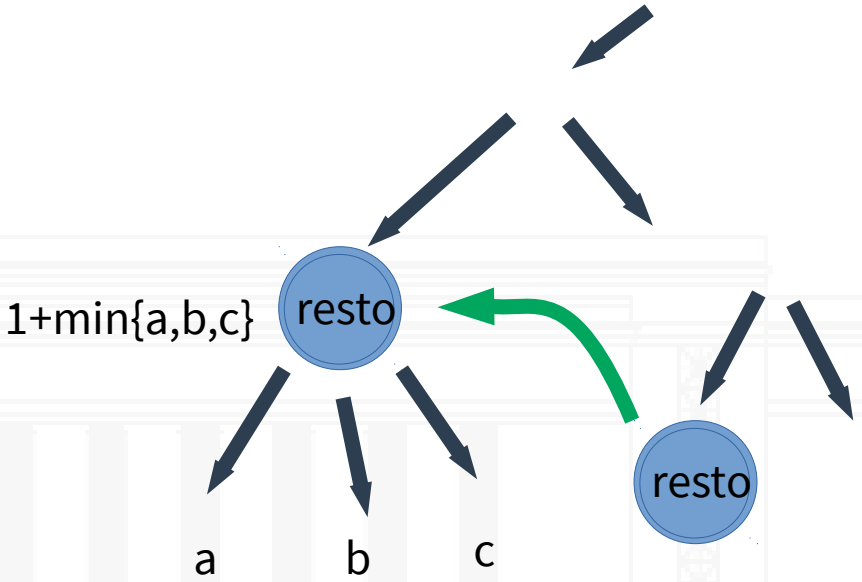
Análisis

Mejoras posibles

Parte de los caminos son iguales

Se los puede calcular solo 1 vez

“es un subproblema”



Subproblema

Calcular el óptimo del cambio X debe usar el mínimo entre los subproblemas X-Cj para j=1...n

Reccurrencia

Podemos expresar el problema como:

$$\begin{cases} OPT(x) = 0 & , si\ x = 0 \\ OPT(x) = 1 + min_{C_i \in \$}\{OPT(x - C_i)\} & , si\ X > 0 \end{cases}$$

El resultado con el mínimo cambio será:

OPT(x)

Se requieren calcular

Los x-1 OPT() anteriores (no hace falta recalcular un valor previamente obtenido)

En cada subproblema se tiene que realizar n comparaciones.

Solución iterativa

Complejidad

Temporal O(n*X)

Espacial: O(X)

Es un algoritmo pseudo-polinómico

```
OPT[0]=0

Desde i=1 a x

     minimo = +∞
     Desde j=1 a n
          resto = i - C[j]
          si resto >= 0 y minimo > OPT[resto]
               minimo = OPT[resto]

     OPT[i] = 1 + minimo

Retornar OPT[x]
```

Reconstruir las elecciones

Para cada subproblema i

almacenar la C_j que retorne el subproblema de mínimo cambio

Reconstruir para atrás

Partiendo de la denominacion de elección para el OPT[X]

Iterar pasando por lo subproblemas elegidos como mínimos

```
OPT[0]=0
elegida[0] = 0
Desde i=1 a x

     minimo = +∞
     elegida[i] = 0
     Desde j=1 a n
          resto = X - C[j]
          si resto >= 0 y minimo > OPT[resto]
               elegida[i] = j
               minimo = OPT[resto]

     OPT[i] = 1 + minimo


resto = x
Mientras resto >0
     Imprimir C[elegida[resto]]
     resto = resto - C[elegida[resto]]

Imprimir OPT[x]
```

.UBAfiuba

FACULTAD DE INGENIERÍA

Presentación realizada en Abril de 2020