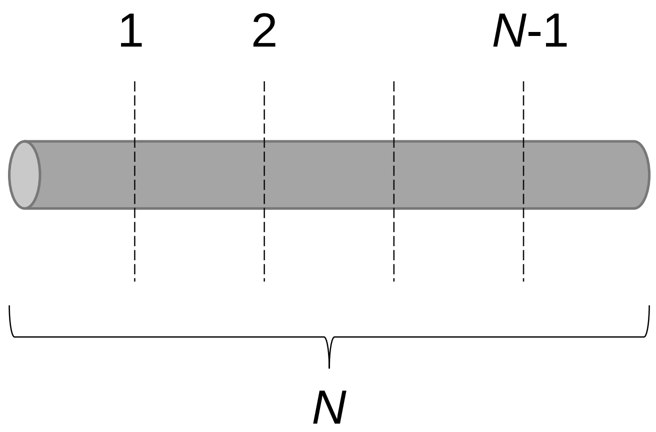
Corte de cable

Corte de cable

Entrada: Un entero *N* correspondiente a la longitud del cable y unos valores no negativos p_k correspondientes al precio de venta de un segmento de longitud k $(1 \leq k \leq N)$



Salida: Esquema de corte que represente la mayor ganancia posible

* La idea es que los precios no son fijos por unidad, ni necesariamente hay economías de escala (en esos casos la solución sería trivial)

Ejemplo: Con *N* = 4

k	1	2	3	4
p	1	5	8	9

Solución: 2 segmentos de longitud 2 cada uno para una ganancia total de 10

Corte de cable

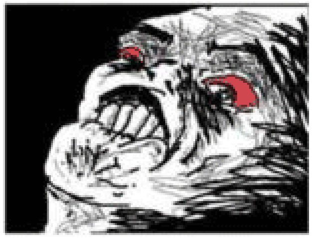
Para resolver mediante programación dinámica ¿Qué forma debería tener la solución óptima de un subproblema?

Dada una longitud *i* (*i* ≤ *N*) la solución óptima para ese subproblema consistiría en elegir una de las siguientes opciones:

1. Cortar un segmento de una unidad (*j*=1) y sumar la mejor ganancia posible del *i-1* restante de cable
2. Cortar un segmento de dos unidades (*j*=*2*) y sumar la mejor ganancia posible del *i-2* restante de cable
3. Cortar un segmento de tres unidades (*j*=*3*) y sumar la mejor ganancia posible del *i-3* restante de cable

…

j. Dejar el segmento de i unidades (*j*=*i*) y no dejar nada restante



Si las conociéramos …

Solución mediante programación dinámica

```
r₁ = p₁
for i = 2 to N:
    rᵢ = pᵢ
    for j = 1 to i-1
        rᵢ = MAX(rᵢ, pⱼ + rᵢ₋ⱼ)
print(r_N)
```

¿Cuál es la eficiencia de este algoritmo?

La cantidad de subproblemas es lineal y cada uno se puede resolver con eficiencia lineal. Esto nos da O($\boldsymbol{N^2}$), sin duda menor que O($\boldsymbol{N2^{N-1}}$) de la solución por búsqueda exhaustiva

Ejemplo: $N = 4$, $p = 1, 5, 8, 9$

r

r	
1	
5	MAX(5, 1+1)
8	MAX(8, 1+5, 5+1)
10	MAX(9, 1+8, 5+5, 8+1)