

Práctico 10 - AYED2

Lautaro Bachmann

Contents

3)		3
	Descripcion	3
	Definicion recursiva	3
4)		4
	Descripcion	4
	Definicion recursiva	4
5)		5
	Descripcion	5
	Definicion recursiva	5
6)		6
	Descripcion	6
	Definicion recursiva	6
7)		7
	Descripcion	7
	Definicion recursiva	7
8) (EN DUDA)		8
	Version 1 (DUDOSA)	8
	Descripcion	8
	Definicion recursiva	8
	Version 2 (MENOS DUDOSA)	9
	Descripcion	9
	Definicion recursiva	9
	Ejemplo	10
	LLamada principal	10
9)		10
	Calcular máximo	10
	Descripcion	10
	Definicion recursiva	10
	Calcular máximo de todo el tablero	10
	Calcular minimo	10
	Descripcion	10
	Definicion recursiva	10
	Calcular minimo de todo el tablero	11

3)

m = [1,2,3,4]

h = [4,3,2,1]

H

Descripcion

harina(i, h) = “El mayor importe posible que se puede obtener haciendo los pedidos de i hasta n, tal que h sea mayor o igual a 0”

Definicion recursiva

$$harina(i, j) = \begin{cases} 0 & j = 0 \\ 0 & i = n, j < h_i \\ m_i & i = n, j \geq h_i \\ harina(i + 1, j) & i > 0, j < h_i \\ \max(& i > 0, j \geq h_i \\ \quad m_i + harina(i + 1, j - h_i), \\ \quad harina(i + 1, j) \\) \end{cases}$$

4)

Descripcion

$globo(i, h)$ = “El menor valor de los objetos que van desde 1 hasta i que tengo que tirar tal que el peso de los objetos tirados sea mayor o igual a h ”

Definicion recursiva

$$globo(i, h) = \begin{cases} 0 & h \leq 0 \\ \infty & i = 0, h > 0 \\ \min(& i > 0, h > 0 \\ & v_i + globo(i-1, h - p_i), \\ & globo(i-1, h) \\) & \end{cases}$$

Veamos un ejemplo:

$p = [1, 2, 3, 4]$

$v = [4, 3, 2, 1]$

```
globo(4, 8)
= min(1 + globo(4-1, 8-4), globo(4-1, h))
= min(1 + min(2 + globo(3-1, 4-3), globo(3-1, 4)),
globo(4-1, 4))
...
```

5)

p = [1,2,3,4]

r = [4,3,5,6]

m = [5,6,7,8]

Descripcion

telefono(d) = “El maximo valor alcanzable alquilando el telefono desde el dia d hasta el ultimo dia”

Definicion recursiva

$$telefono(d) = \begin{cases} 0 & \forall i : p_i > d \\ \max(& else \\ telefono(d+1), & \\ \max_{\{i \text{ tal que } p_i = d\}} (m_i * (r_i - p_i + 1) + telefono(r_i + 1)) & \\) & \end{cases}$$

6)

Descripcion

$\text{prima}(i, A, B)$ = “Calcula el maximo valor venta alcanzable para cada objeto de 1 hasta i tal que la cantidad de material A y la cantidad de material B sean mayores o iguales a 0”

Definicion recursiva

$$\text{prima}(i, A, B) = \begin{cases} 0 & A = 0 \vee B = 0 \\ 0 & i = 0 \\ \text{prima}(i-1, A, B) & A < a_i \vee B < b_i \\ \max(& A \geq a_i \wedge B \geq b_i \\ \quad v_i + \text{prima}(i-1, A - a_i, B - b_i), \\ \quad \text{prima}(i-1, A, B) \\) \end{cases}$$

7)

Descripcion

$mochilas(i, w1, w2)$ = “Calcula el maximo valor valor que se puede alcanzar cargando objetos en las mochilas $w1$ y $w2$ tal que el peso de los objetos no supere al peso soportado por ambas mochilas”

Definicion recursiva

$$mochilas(i, w1, w2) = \begin{cases} 0 & i = 0 \\ 0 & w1 = 0 \wedge w2 = 0 \\ // \text{ no entra en ninguna} \\ mochilas(i-1, w1, w2) & w_i > w1 \wedge w_i > w2 \\ // \text{ entra en primera} \\ \max(& w_i \leq w1 \wedge w_i > w2 \\ \quad v_i + mochilas(i-1, w1 - w_i, w2), \\ \quad mochilas(i-1, w1, w2) \\) & \\ // \text{ entra en segunda} \\ \max(& w_i > w1 \wedge w_i \leq w2 \\ \quad v_i + mochilas(i-1, w1, w2 - w_i), \\ \quad mochilas(i-1, w1, w2) \\) & \\ // \text{ entra en ambas} \\ \max(& w_i \leq w2 \wedge w_i \leq w2 \\ \quad v_i + mochilas(i-1, w1 - w_i, w2), \\ \quad v_i + mochilas(i-1, w1, w2 - w_i), \\ \quad mochilas(i-1, w1, w2) \\) & \end{cases}$$

8) (EN DUDA)

Version 1 (DUDOSA)

Descripcion

$automoviles(j)$ = “Encuentra el costo minimo de fabricar un automovil desde la estacion de ensamble numero j hasta el inicio de la linea de ensamble utilizando ambas lineas de ensamble disponibles”

Definicion recursiva

$$automoviles(j) = \begin{cases} 0 & j = 0 \\ \text{Min}_{i \in \{1,2\}} (& j > 0 \\ \quad a_{i,j} + automoviles(j-1) & \\ \quad \mathbf{min} \ a_{i,j} + t_{i,j} + automoviles(j-1) & \\) & \end{cases}$$

Version 2 (MENOS DUDOSA)

Descripcion

$automoviles(i,j)$ = “Encuentra el costo minimo de fabricar un automovil estando en la linea de ensamble S_i desde la estacion de ensamble numero j hasta la estacion de ensamble n ”

Definicion recursiva

$$automoviles(i,j) = \begin{cases} a_{i,j} & j = n \\ \min(& i = 1 \\ & a_{i,j} + automoviles(i,j+1), \\ & a_{i,j} + t_{i,j} + automoviles(i+1,j+1) \\) & \\ \min(& i = 2 \\ & a_{i,j} + automoviles(i,j+1), \\ & a_{i,j} + t_{i,j} + automoviles(i-1,j+1) \\) & \end{cases}$$

Ejemplo

Llamada principal

```
r:= max(automoviles(1,1), automoviles(2,1))
```

9)

Calcular máximo

Descripcion

$\text{maxUp}(i,j)$ = “Encuentra el maximo puntaje que se puede obtener partiendo de la posicion i,j , siendo i la fila y j la columna”

Definicion recursiva

$$\text{maxUp}(i,j) = \begin{cases} c_{i,j} & i = n \\ \max(& i < n \\ \quad c_{i,j} + \text{maxUp}(i+1, \max(0, j-1)), & // \text{ Arriba a la izq} \\ \quad c_{i,j} + \text{maxUp}(i+1, j), & // \text{ Arriba al centro} \\ \quad c_{i,j} + \text{maxUp}(i+1, \min(j+1, n)) & // \text{ Arriba a la der} \\) \end{cases}$$

Calcular máximo de todo el tablero

```
var maximo: nat
maximo:= 0
for j := 1 to n do
  maximo:= max(maximo, maxUp(0,j))
od
```

Calcular minimo

Descripcion

$\text{maxUp}(i,j)$ = “Encuentra el minimo puntaje que se puede obtener partiendo de la posicion i,j , siendo i la fila y j la columna”

Definicion recursiva

$$\text{minUp}(i,j) = \begin{cases} c_{i,j} & i = n \\ \min(& i < n \\ \quad c_{i,j} + \text{minUp}(i+1, \min(0, j-1)), & // \text{ Arriba a la izq} \\ \quad c_{i,j} + \text{minUp}(i+1, j), & // \text{ Arriba al centro} \\ \quad c_{i,j} + \text{minUp}(i+1, \min(j+1, n)) & // \text{ Arriba a la der} \\) \end{cases}$$

Calcular minimo de todo el tablero

```
var minimo: nat
minimo:=  $\infty$ 
for j := 1 to n do
  minimo:= min(minimo, minUp(0,j))
od
```