Informática
Año 2013/2014
Facultad de CC. Matemáticas

Hoja de ejercicios 6

Recursión

3 de abril de 2014

▷ 1. Funciones recursivas simples

Escribe funciones recursivas para calcular:

- El máximo común divisor de dos enteros.
- El término n-ésimo de la sucesión de Fibonacci.
- El factorial de un número natural.

▷ 2. Las torres de Hanoi

> *"En el gran templo de Benarés, bajo la cúpula que marca el centro del mundo, descansa una base de bronce, sobre la que hay fijadas tres agujas de diamante, cada una del grueso del cuerpo de una abeja y la altura de un codo. En una de esas agujas, en el instante de la Creación, Dios colocó sesenta y cuatro discos de oro, el mayor sobre el plato de bronce, y el resto en orden decreciente de tamaños. Ésa es la Torre de Bramah. Día y noche, incesantemente, los sacerdotes del templo mueven los discos de una aguja de diamante a otra de acuerdo con las inmutables leyes de Bramah, que requieren que el sacerdote a cargo de la tarea no mueva más de un disco a la vez, y que al mover un disco lo coloque sin dejar debajo ninguno de menor tamaño. Cuando los sesenta y cuatro discos hayan sido así transferidos de la aguja en la que colocó Dios los discos en el instante de la Creación a otra de las agujas, la torre, el templo y los bramahnes se convertirán en polvo, y con un trueno el mundo desaparecerá."* (De Parville, 1884)

Haz un programa que, partiendo de una posición inicial con n discos como la de la figura 1, transfiriendo los discos según las leyes de Bramah, llegue a la posición en la que todos los discos están en la aguja C. El programa debe volcar en un archivo la representación de las sucesivas configuraciones por las que van pasando las agujas, desde la posición inicial a la final, debidamente numeradas, con el formato del Ejemplo. El número de discos n debe ser una constante.

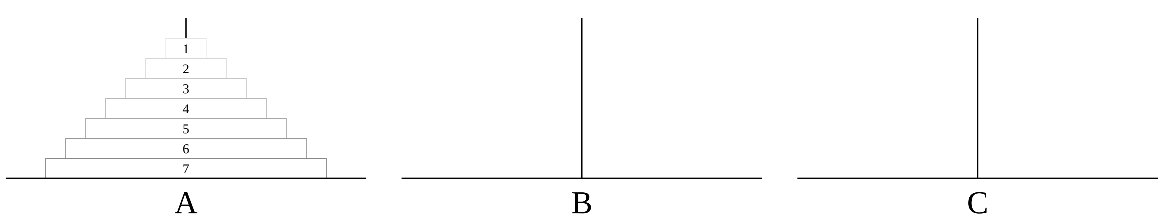

Figura 1: Instante de la creación, con $n = 7$ discos.

Para encontrar la estrategia a seguir conviene plantearse lo siguiente. En algún momento habrá que mover el disco número n, el mayor de todos. Si pretendemos moverlo a C, por fuerza C debe estar vacía, de modo que necesariamente todos los demás discos están en B, y debidamente ordenados, exactamente como en la figura 2.

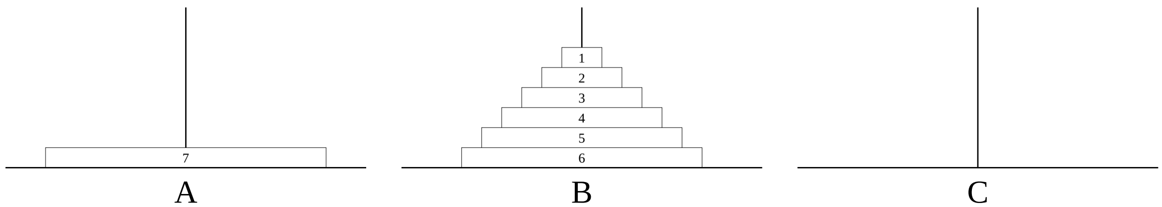

Figura 2: Única situación en la que es posible mover el disco mayor de A a C.

Es fácil extraer de este caso la idea general, que permite resolver de forma simple y elegante el problema: La transferencia de los m discos superiores de una aguja A_1 a otra A_2 usando como auxiliar A_3 puede dividirse en tres fases:

1. Transferencia de $m - 1$ discos de A_1 a A_3 usando A_2 como auxiliar.

2. Paso de un disco de A_1 a A_2.

3. Transferencia de $m - 1$ discos de A_3 a A_2 usando A_1 como auxiliar.

Ejemplo En la figura 3 se transcribe el principio del archivo generado para el caso $n = 4$.
matematicas.discreta.combinatoria.num-comb

▷ 3. Números combinatorios: funciones sencillas

En este ejercicio se proponen diferentes maneras de hallar y usar números combinatorios. Dados los enteros positivos m y n, tales que $m \geq n$, se pide escribir funciones para hallar $\binom{m}{n}$, de diversos modos. Para cada una de las funciones desarrolladas, repasa cuidadosamente si se debe exigir algún requisito (precondición) sobre los datos, e indícalo mediante un comentario adecuado.

A partir del factorial Mediante la fórmula $\frac{m!}{(m-n)!n!}$, incluyendo el `factorial` como una función local.

Versión recursiva trivial Mediante una función recursiva, recordando que

$$\binom{m}{0} = \binom{m}{m} = 1$$

$$\binom{m}{n} = \binom{m-1}{n} + \binom{m-1}{n-1}, \quad \text{si} \quad 1 \leq n \leq m.$$

Otra versión recursiva Usando la relación de recurrencia

$$\binom{m}{n} = \frac{m}{n}\binom{m-1}{n-1}, \quad \text{siendo} \quad \binom{m}{0} = 1.$$

Una pequeña mejora más Incluyendo en las versiones anteriores el hecho de que $\binom{m}{n} = \binom{m}{m-n}$ para economizar cálculos ... cuando convenga.

```
Configuración 0:
------------------

   11
  2222
 333333
44444444
--------
 Torre 1



--------
 Torre 2



--------
 Torre 3

Configuración 1:
------------------

  2222
 333333
44444444
--------
 Torre 1



   11
--------
 Torre 2



--------
 Torre 3
```

Figura 3: Principio del fichero generado para $n = 4$.

▷ 4. Descomposición en factores primos

Seguro que has realizado alguna vez la tediosa tarea de descomponer un número en factores primos:

$$
\begin{array}{r|l} 12 & 2 \\ 6 & 2 \\ 3 & 3 \\ 1 & \end{array}
\qquad
\begin{array}{r|l} 85 & 5 \\ 17 & 17 \\ 1 & \end{array}
\qquad
\begin{array}{r|l} 56 & 2 \\ 28 & 2 \\ 14 & 2 \\ 7 & 7 \\ 1 & \end{array}
\qquad
\begin{array}{r|l} 110 & 2 \\ 55 & 5 \\ 11 & 11 \\ 1 & \end{array}
$$

¿Podríamos hacer un programa que fuése capaz de hacer estas descomposiciones? Seguro que sí. Escribe un programa que reciba como entrada un número entero positivo y calcule la descomposición del mismo en factores primos.

Ejemplo Un ejemplo de ejecución podría ser el siguiente:

```
Numero a descomponer? -1
El número ha de ser positivo
Numero a descomponer? 12
Factores primos: 2 2 3
Otro número (s/n)? s

Numero a descomponer? 85
Factores primos: 5 17
Otro número (s/n)? s

Numero a descomponer? 56
Factores primos: 2 2 2 7
Otro número (s/n)? s

Numero a descomponer? 110
Factores primos: 2 5 11
Otro número (s/n)? n
```

▷ 5. Descomposiciones de un natural

Nos preocupamos por las posibles formas de descomponer un número natural como suma de naturales mayores que cero.

- Escribe una función que, dados un entero n y una lista de listas l, devuelva la lista que se obtiene al añadir n al principio de cada lista de l. El subprograma no debe modificar l.

  Si $n = 5$ y $l = [[2, 3, 4], [3, 3, 3]]$ la función debe devolver $[[5, 2, 3, 4], [5, 3, 3, 3]]$

- Escribe una función que ,dado un número natural n, devuelva la lista de las posibles descomposiciones de n como suma de números naturales mayores que cero, representaremos cada descomposición como una lista de naturales constituida por los correspondientes sumandos.

  Por ejemplo:

  - Si $n = 1$, la función debe devolver $[[1]]$

- Si $n = 2$, la función debe devolver $[[1, 1], [2]]$
- Si $n = 3$, la función debe devolver $[[1, 1, 1], [1, 2], [2, 1], [3]]$

Si piensas el ejercicio de forma recursiva debes comenzar por plantearte que debe devolver la función si $n = 0$.

- Queremos eliminar las redundancias del estilo $3 = 1 + 2 = 2 + 1$, ¿puedes definir la función anterior de forma que las descomposiciones que son iguales salvo el orden no se repitan?

▷ **6. Coordenadas naturales**

Sean n un número natural, $\overrightarrow{v}$ un vector de enteros positivos y $\overrightarrow{c}$ un vector de naturales. Se dice que $\overrightarrow{c}$ es un vector de coordenadas de n en el sistema $\overrightarrow{v}$ si $\overrightarrow{c} \cdot \overrightarrow{v} = n$, es decir, $\sum_i c_i \cdot v_i = n$.

Implementa en Python una función `expresable`, que en caso de que exista un vector $\overrightarrow{c}$ de coordenadas para n en el sistema $\overrightarrow{v}$, lo almacena en el parámetro de salida `c`, devolviendo `true`. En otro caso, devuelve `false`.