

Trabajo Práctico 5: Naturales

Programación I

El objetivo de este trabajo práctico es diseñar programas recursivos sobre números naturales.

1 Enunciado

Ejercicio 1. Diseñe una función `subir` que dado un número natural n , determine de cuántas formas distintas se puede subir una escalera de n escalones si se suben los escalones, únicamente, de a uno, de a tres o de a cinco.

Por ejemplo, hay tres formas distintas de subir una escalera de cuatro escalones:

- subir un escalón, subir un escalón, subir un escalón, subir un escalón;
- subir tres escalones, subir un escalón;
- subir un escalón, subir tres escalones.

Observar que no es posible subir escalones de a cinco en este caso, ya que la escalera tiene menos de cinco escalones.

Ayuda. Imagine que una persona ha subido la escalera siguiendo las restricciones que se indican en el enunciado. ¿Cuál pudo haber sido el escalón que pisó justo antes de pisar el último escalón?.

Ejercicio 2. Se tiene un tablero de m filas y n columnas, con un número entero en cada casilla. La casilla (i, j) corresponde a la i -ésima columna y j -ésima fila, donde $0 \leq i < n$ y $0 \leq j < m$.

Un *recorrido válido* del tablero es una secuencia de $n+m-1$ casillas del mismo de manera tal que:

- la primer casilla es la casilla $(0,0)$, es decir la casilla superior izquierda del tablero,
- la última casilla es la casilla $(n-1, m-1)$, es decir la casilla inferior derecha del tablero,
- cada casilla en la secuencia es la inmediata inferior o a la inmediata a la derecha de la anterior.

Por ejemplo, el siguiente tablero de 3 filas y 2 columnas

-5	10
0	-2
9	3

tiene tres recorridos válidos:

-5	10
0	-2
9	3

-5	10
0	-2
9	3

-5	10
0	-2
9	3

Definimos el *costo* de un recorrido como la suma de los números correspondientes a las casillas que lo integran.

En el ejemplo anterior, el recorrido de la izquierda tiene costo 7, mientras que el central tiene costo -4 y el de la derecha tiene costo 6.

Dado un tablero con las características descritas, se pide calcular el máximo costo que puede tener un recorrido válido.

En el ejemplo dado, el máximo costo de un recorrido válido es 7 (que corresponde al recorrido de la izquierda).

Aunque pueda no resultarle intuitivo, en algunas ocasiones, para resolver un problema es conveniente plantear una versión más general del mismo. Una vez resuelta esa generalización, la respuesta al problema original se obtiene fácilmente (ya que constituye, simplemente, una instancia particular del problema general).

Podemos ver que ese patrón se aplica en este caso. En lugar de calcular directamente el costo máximo de un recorrido válido desde la casilla superior izquierda a la casilla inferior derecha, podemos calcular el costo máximo de un recorrido válido desde la casilla superior izquierda a otra casilla (i, j) , donde $0 \leq i < n$ y $0 \leq j < m$. De esta manera, resolveremos nuestro problema original instanciando esta versión general en $i = n-1$ y $j = m-1$.

¿Por qué es más conveniente expresar nuestro problema de esta forma? Básicamente, porque para resolver problemas “grandes” suele ser conveniente dividirlos en problemas “más pequeños”, resolver esos problemas más chicos y finalmente combinar sus soluciones para obtener una solución al problema “más grande”. Esta técnica se conoce como *Divide & Conquer*.

Volviendo al problema en cuestión, observamos que, cuando $i, j > 0$, es fácil calcular el costo máximo de un recorrido válido de $(0,0)$ a (i, j) si sabemos cuál es el costo máximo de un recorrido válido de $(0,0)$ a $(i-1, j)$ y de $(0,0)$ a $(i, j-1)$. ¿Por qué?.

1. Con el objetivo de ayudarla/o a pensar una regla general para resolver el problema en cuestión, se pide que, usando lápiz y papel, identifique cada uno de los diez recorridos válidos del siguiente tablero y calcule el costo de cada uno de ellos.

5	4	-10	15
1	-2	8	2
-22	9	1	4

2. Notamos con $tab[i][j]$ al valor de la casilla de la i -ésima columna y j -ésima fila de un tablero tab .

Complete la siguiente fórmula para calcular el costo máximo de un recorrido de $(0, 0)$ a (i, j) en el tablero tab :

$$maximoCosto(i, j) = \begin{cases} tab[0][0] & \text{si } i = j = 0 \\ \dots & \text{si } i = 0 \text{ y } j \neq 0 \\ \dots & \text{si } j = 0 \text{ e } i \neq 0 \\ \dots & \text{en otro caso} \end{cases}$$

Observamos que esta es una fórmula recursiva, pero a diferencia de las vistas en la práctica, se hace recursión sobre dos argumentos en lugar de uno.

3. Para representar tableros, usaremos listas de listas de enteros. Representaremos a un tablero de m filas y n columnas como una lista de m listas de enteros, cada una de tamaño n , representando a cada fila.

Para representar el tablero del ejemplo, usamos el siguiente código:

```
⋮ (define TABLERO (list (list -5 10) (list 0 -2) (list 9 3)))
```

Diseñe una función `maximo-costo` que dados una lista de listas de enteros tab , que representa un tablero de m filas y n columnas, y dos naturales i y j devuelva el máximo costo de un recorrido válido de $(0,0)$ a (i,j) en el tablero tab . Puede asumir que $0 \leq i < n$, $0 \leq j < m$.

Ayuda. Siga la fórmula del apartado anterior.

4. Diseñe una función `maximo-costo-tablero` que dada una lista de listas de enteros tab , que representa un tablero de m filas y n columnas, devuelva el máximo costo que puede tener un recorrido válido en tab .

2 Sugerencias

- Recuerde que el primer paso en la resolución de problemas consiste en el análisis del problema a resolver. Para ello es esencial que lea atentamente el enunciado las veces que sean necesarias para asegurarse de haber entendido correctamente qué se pide. Resulta útil usar lápiz y papel para razonar sobre instancias pequeñas del problema.
- No intente escribir código sin antes haber pensado la solución usando lápiz y papel. En general, es más difícil modelar el problema y encontrar una solución que llevar esa solución a *racket*.

3 Formato de entrega

- El trabajo práctico debe resolverse en grupos de dos integrantes. Cada persona puede participar en un único grupo. Un grupo puede estar integrado por personas de distinta comisión.
- Exactamente una persona por grupo debe realizar la entrega en el sitio, escribiendo en el campo "Comentarios" los apellidos y nombres de cada integrante.

- No se aceptarán entregas en las que no se haya seguido la receta para el diseño. Para que la entrega del trabajo práctico sea válida, todas las funciones deben contar con diseño de datos, signatura, declaración de propósito, casos de prueba (si corresponde) y código. En caso contrario, se considerará que el TP no fue entregado.
- La entrega consiste en un único archivo por grupo: el archivo TP5-Apellido1-Apellido2.rkt que deberá ser editado apropiadamente. En concreto, se solicita:
 - Completar los datos pedidos para cada integrante del grupo según se indica en el archivo.
 - Completar el archivo con la resolución.
 - Cambiar el nombre del archivo reemplazando Apellido1 y Apellido2 por los apellidos de los integrantes, en orden alfabético ascendente.

En el ejercicio 3-1, no hace falta que muestre los distintos recorridos, sólo indique cuáles son los 10 costos.