**Evaluación perezosa**

¿QUÉ ES LA EVALUACIÓN PEREZOSA? Evaluación impaciente (eager): el evaluador hace todo lo que puede. Corresponde a llamada por-valor. Evaluación perezosa (lazy):

• El evaluador hace solamente lo preciso. Corresponde a llamada pornecesidad.

• Significa: Haz sólo lo que te pida un patrón a la izquierda de una ecuación o cualificador (where o let).

• Es una estrategia de evaluación que retrasa la evaluación de una expresión hasta que el valor de esto realmente se requiera (evaluación no estricta) y que también evita evaluaciones repetidas (compartimiento de ciencias informáticas). El compartimiento puede reducir la duración de ciertas funciones por un factor exponencial sobre otras estrategias de evaluación no estrictas, como la llamada de nombre. Las ventajas de la evaluación perezosa incluyen:

• El rendimiento aumenta debido a evitación de cálculos innecesarios y evitación de condiciones de error en la evaluación de expresiones compuestas.

• La capacidad de construir estructura de datos potencialmente infinita

• La capacidad de definir estructura de control como abstracciones en vez de como obras primitivistas.

La evaluación perezosa puede llevar a la reducción de la huella de memoria, ya que los valores se crean cuando necesario. Sin embargo, con la evaluación perezosa, es difícil combinarse con rasgos imperativos como la excepción que se maneja (manejo de la excepción) y entrada/salida (entrada/salida), porque el pedido de operaciones se hace indeterminado. La evaluación perezosa puede introducir el agujero espacial.

También, la depuración es difícil. ESTRATEGIAS DE PROGRAMACIÓN PEREZOSA Para los ejemplos se considera la función mult :: (Int,Int) ◊ Int mult (x,y) = x \* y

• Evaluación mediante paso de parámetros por valor (o por más internos): mult (1+2,2+3) = mult (3,5) [por def. de +] = 3\*5 [por def. de mult] = 15 [por def. de \*]

• Evaluación mediante paso de parámetros por nombre (o por más externos): mult (1+2,2+3)= (1+2)\*(3+5) [por def. de mult] = 3\*5 [por def. de +]

• Evaluación con lambda expresiones mult’ (1+2) (2+3) = mult’ 3 (2+3) [por def. de +] = (λy → 3\*y) (2+3) [por def. de mult’] = (λy → 3\*y) 5 [por def. de +] = 3\*5 [por def. de +] = 15 [por def. de \*]