UNIVERSIDAD
DE SANTIAGO
DE CHILE

Recursión

Métodos de Programación

1-2020

UNIVERSIDAD DE SANTIAGO DE CHILE

CONTENIDO

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

UNIVERSIDAD DE SANTIAGO DE CHILE

INTRODUCCIÓN

- ¿Qué elementos se vieron la clase pasada?
- ¿Para qué servía cada uno?
- Hoy aprenderemos algo nuevo, llamado recursión.
- Será una nueva forma de enfrentarse a los problemas.

UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [1/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- La recursión es una técnica para resolver problemas.
- Es utilizada para repetir procesos hasta llegar a un resultado conocido.
- La recursión es descrita como una función que se utiliza a si misma para poder resolver el problema.
  - $Función(x) = Función(x')$

UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [2/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- Las funciones recursivas poseen dos elementos para poder definirlas:
  - **Caso Base:**
    - Consiste en uno o varios valores conocidos, es decir, que para el valor de entrada de la función, se sabe cual es su resultado.
  - **Llamada recursiva:**
    - Consiste en la llamada a la misma función, pero con un parámetro de **entrada actual** distinto **al formal**, con el fin de que este valor se acerque al caso base.

UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [3/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- Un ejemplo de una función recursiva es la sumatoria de los n primeros números naturales.

- $\sum_{i=1}^{n} i = \begin{cases} n + \sum_{i=1}^{n-1} i \; ; Si \; n > 1 \\ 1 ; Si \; n = 1 \end{cases}$

Función a definir

Caso conocido, o caso base

Definición de la función llamándose a si misma.

Nota: el valor de n, cambio por n-1, de esta forma se asegura para llegar al valor de 1, independiente del valor del número natural ingresado.

UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [4/10]

- Pasando a pseudo código la función anterior, se tiene lo siguiente:

Entrada: un número natural (número entero mayor que 0)

Salida: Un número natural

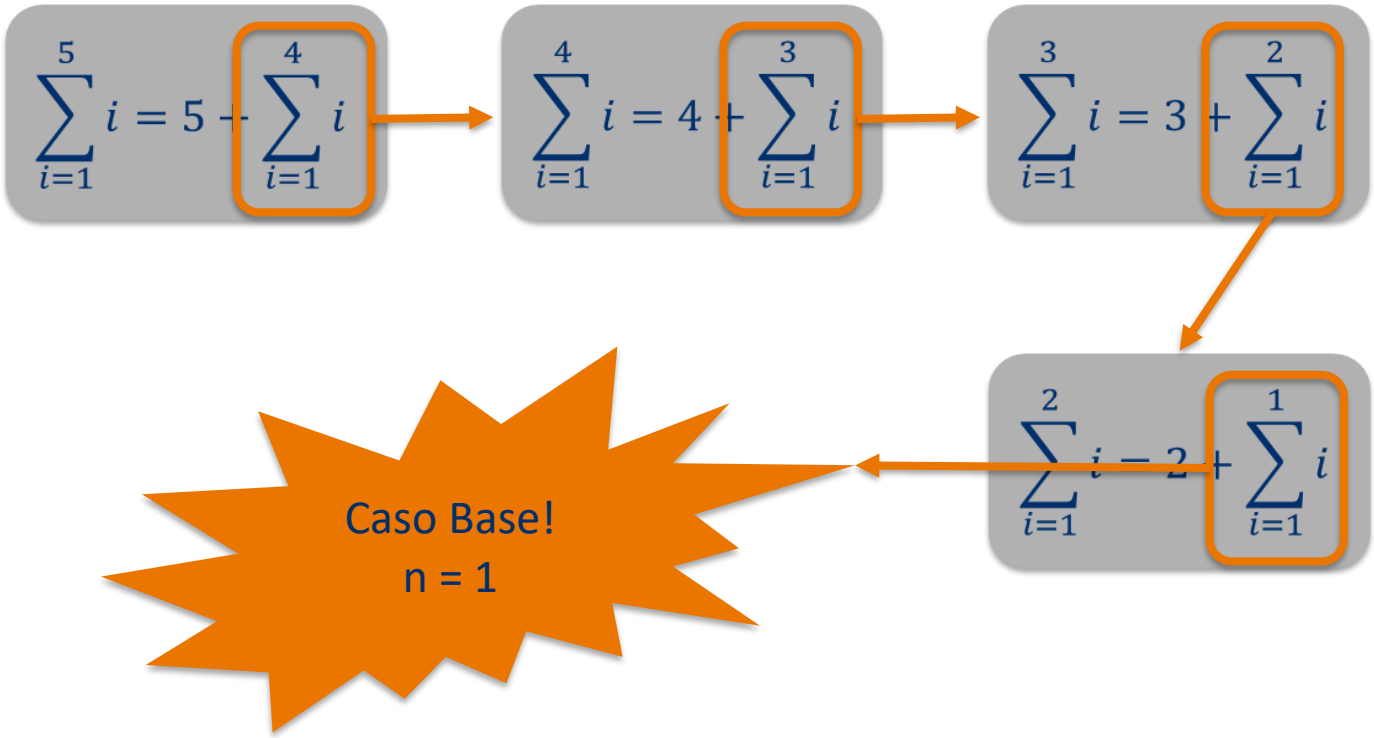
```
1-. Define Función sumatoria(n):
        1.1-. Si n es igual a 1:
                1.1.1-. devolver 1
        1.2-. Sino:
                1.2.1-. devolver n + sumatoria(n-1)
```

UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [5/10]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

- ¿Cómo se iría resolviendo este problema?
  - Consideremos un n = 5

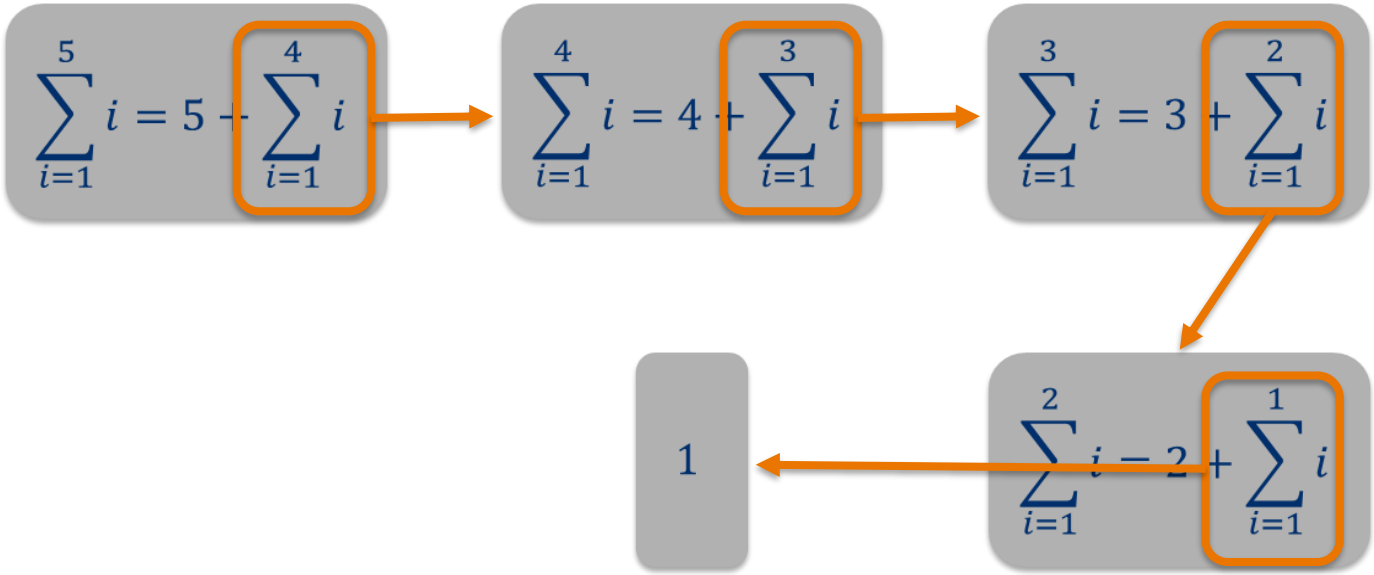


UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [6/10]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

- ¿Cómo se iría resolviendo este problema?
  - Consideremos un n = 5

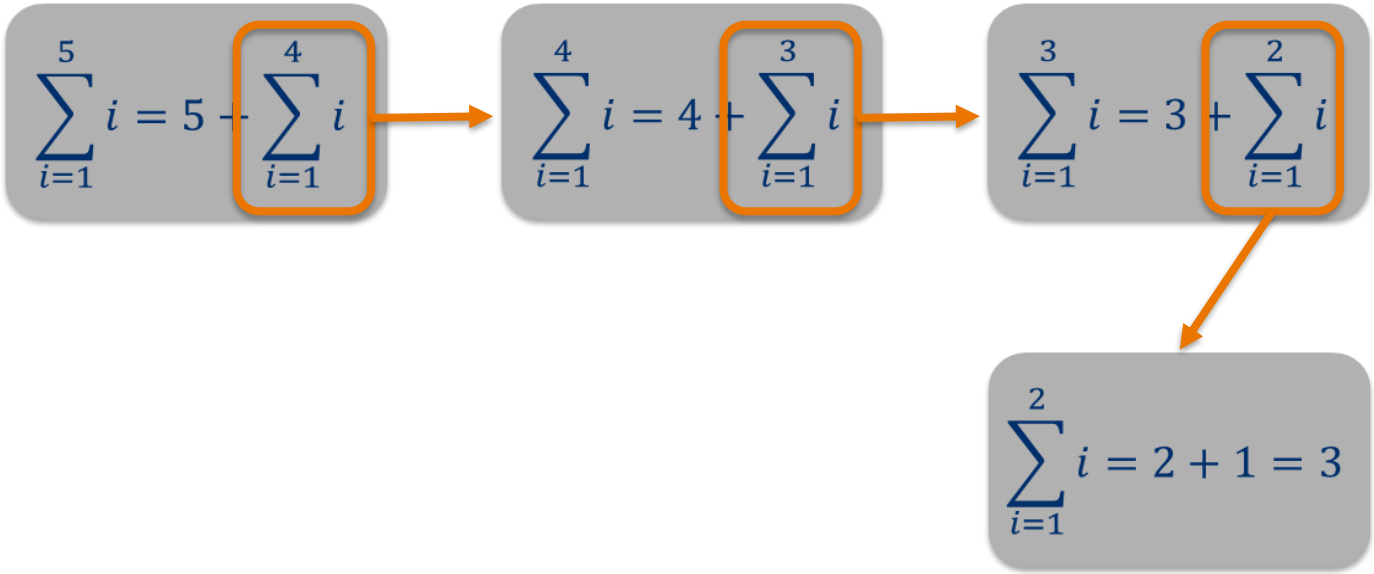


UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [7/10]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

- ¿Cómo se iría resolviendo este problema?
  - Consideremos un n = 5

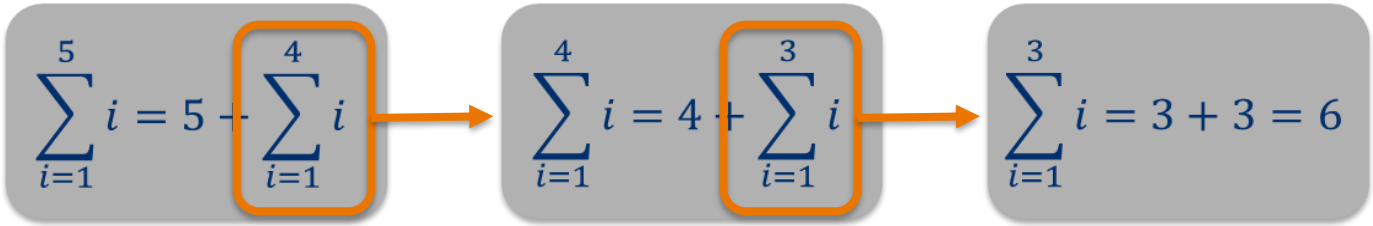


UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [8/10]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

- ¿Cómo se iría resolviendo este problema?
  - Consideremos un n = 5

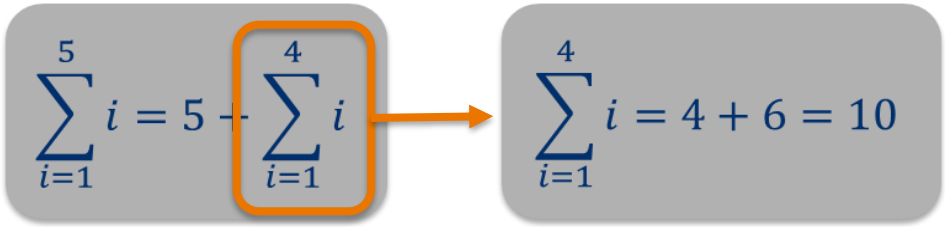


UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [9/10]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

- ¿Cómo se iría resolviendo este problema?
  - Consideremos un n = 5



UNIVERSIDAD DE SANTIAGO DE CHILE

CONCEPTO RECURSIÓN [10/10]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

- ¿Cómo se iría resolviendo este problema?
  - Consideremos un n = 5

$$\sum_{i=1}^{5} i = 5 + 10 = 15$$

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [1/10]

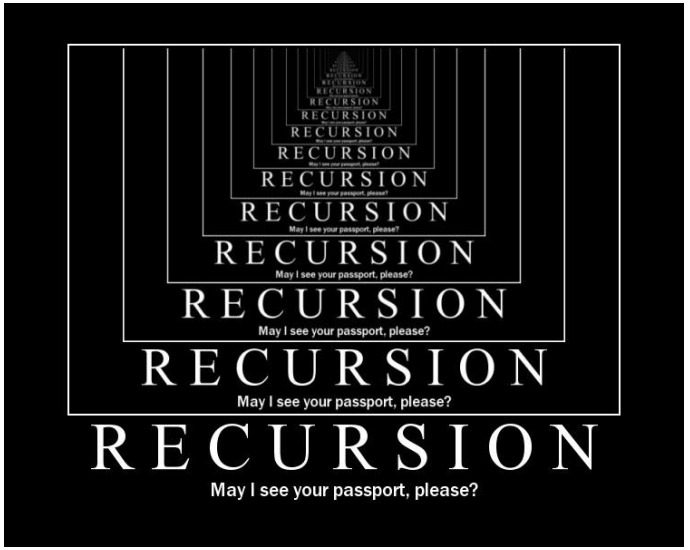
Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- A pesar que el concepto de recursión en general es el mismo, donde una función se llama así misma, cambiando la entrada para llegar a un caso conocido, se pueden clasificar en distintos tipos:
  - Simple y múltiple.
  - Directa e indirecta.
  - Anidada y no.
  - De cola



Obed_LLP - blogger

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [2/10]

- La **recursión simple** es aquella que posee solo una llamada recursiva en su definición.
  - $\sum_{i=1}^{n} i \begin{cases} 1, Si\ n\ es\ 1 \\ n + \sum_{i=1}^{n-1} i, Si\ n \geq 1 \end{cases}$
- La **recursión múltiple** es aquella que posee más de una llamada recursiva en su definición.
  - $fibonacci(n) = \begin{cases} 1, Si\ n = 0 \\ 1, Si\ n = 1 \\ fibonacci(n-1) + fibonacci(n-2), Si\ n > 1 \end{cases}$

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [3/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- La **recursión directa** es aquella que se llama a si mismo dentro de la llamada recursiva.

  – $factorial(n) = \begin{cases} 1; Si\ n = 0 \\ 1; Si\ n = 1 \\ n * factorial\ (n-1); Si\ n > 1 \end{cases}$

- La **recursión indirecta** es aquella que posee una llamada recursiva a otra función recursiva, la cual llama a la primera.

$$par(n) = \begin{cases} Verdadero, ; Si\ n = 0 \\ Falso; Si\ n = 1 \\ impar(n-1); Si\ n > 1 \end{cases} \qquad impar(n) = \begin{cases} Falso\ ; Si\ n = 0 \\ Verdadero\ ; Si\ n = 1 \\ par(n-1); Si\ n > 1 \end{cases}$$

TIPOS RECURSIÓN [4/10]

- La **recursión anidada** es aquella que posee una llamada recursiva, dentro de su misma llamada recursiva.
  - *Ackermann*
  - $$Ack(m,n) = \begin{cases} n+1; Si\ m = 0 \\ Ack(M-1,1); Si\ m > 0\ y\ n = 0 \\ Ack\big(m-1, Ack(mn-1)\big); Si\ m > 0\ y\ n > 0 \end{cases}$$

Values of A(*m*, *n*)

m\n	0	1	2	3	4	n
0	1	2	3	4	5	$n+1$
1	2	3	4	5	6	$n+2 = 2+(n+3)-3$
2	3	5	7	9	11	$2n+3 = 2\cdot(n+3)-3$
3	5	13	29	61	125	$2^{(n+3)}-3$
4	13 $=2^{2^2}-3$	65533 $=2^{2^{2^2}}-3$	$2^{65536}-3$ $=2^{2^{2^{2^2}}}-3$	$2^{2^{65536}}-3$ $=2^{2^{2^{2^{2^2}}}}-3$	$2^{2^{2^{65536}}}-3$ $=2^{2^{2^{2^{2^{2^2}}}}}-3$	$\underbrace{2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}}_{n+3}-3$

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [5/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- La **recursión de cola (Tail rcursion)**
- Primero hay que entender que los llamados recursivos se van acumulando.
- Por ejemplo, un pseudo código de la función factorial es:

Entrada: un número natural

Salida: Un número natural

1-. Define Función factorial(n):

1.1-. Si n es igual a 1 o igual a 0:

1.1.1-. devolver 1

1.2-. Sino:

1.2.1-. devolver n * factorial(n-1)

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [6/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- Al resolver los casos de n > 1, se tiene la instrucción n * factorial(n-1)
- La multiplicación no se realiza directamente, ya que hace falta el siguiente operador, por lo tanto se debe guardar la operación para hacerse posteriormente.
- Esto genera que las llamadas recursivas queden en un estado pendiente, cosa que le pasa a la mayoría de llamadas recursivas, quedando en una **PILA** de estados.

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [7/10]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

Pila de estados pendientes
Factorial(1) = 1
2*Factorial(1)
3*Factorial(2)
4*Factorial(3)
Factorial(4)

- Cuando la llamada es muy grande (el n es muy mayor) puede que el computador no posea memoria suficiente para almacenar todos los estados pendientes.

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [8/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- En esos momentos es necesario utilizar la recursión de **COLA**, la cual trata de evitar esos estados pendientes, resolviéndolos en la misma llamada.

- Utilizando recursión de cola, nuestro pseudo código quedaría de la siguiente forma:

Entrada: dos números naturales
Salida: Un número natural

1-. Define Función factorialDeCola(n, resultado):
    1.1-. Si n es igual a 1 o igual a 0:
        1.1.1-. devolver resultado
    1.2-. Sino:
        1.2.1-. devolver factorialDeCola(n-1, n*resultado)

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [9/10]

- Pero cuando queremos saber el valor del factorial de un número, lo solicitamos con un solo valor, por lo cual deberemos definir una función previa que haga el llamado al factorial de cola.

Entrada: un número natural

Salida: Un número natural

1-. Define Función factorial (n):

1.1-. devolver factorialDeCola(n, 1)

UNIVERSIDAD DE SANTIAGO DE CHILE

TIPOS RECURSIÓN [10/10]

Introducción

Concepto Recursión

Tipos de recursión

Comparación con la iteración

- Con esto ya no existe una pila de estados pendientes, dado que no hay operaciones pendientes antes de la llamada recursiva.
- Es posible pasar una recursión normal (o de pila) a una de cola, siempre y cuando, la llamada recursiva sea lo último que se haga.

UNIVERSIDAD DE SANTIAGO DE CHILE

COMPARACIÓN CON LA ITERACIÓN [1/3]

- Las funciones recursivas también son posibles de representarlas de forma iterativa.
- ¿Pero porqué utilizar recursión o iteración?
- Comparemos dos algoritmos, que hagan lo mismo, de forma iterativa y recursiva.

UNIVERSIDAD DE SANTIAGO DE CHILE

COMPARACIÓN CON LA ITERACIÓN [2/3]

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

Entrada: un número natural

Salida: Un número natural

```
1-. Define Función factorial(n):
        1.1-. Si n es igual a 1 o igual a 0:
                1.1.1-. devolver 1
        1.2-. Sino:
                1.2.1-. devolver n * factorial(n-1)
```

```
1-. Define Función factorial(n):
        1.1-. Define resultado = 1
        1.2-. Mientras n > 0, hacer:
                1.2.1-. resultado = resultado * n
                1.2.1-. n = n – 1
        1.2-. Devolver resultado
```

UNIVERSIDAD DE SANTIAGO DE CHILE

COMPARACIÓN CON LA ITERACIÓN [3/3]

- Ventajas de la recursión:
  - Su algoritmo es más entendible que los iterativos.
  - Los códigos normalmente son más cortos.
- Desventajas de la recursión:
  - Se deben crear muchos estados pendientes.
  - Se puede utilizar mucha memoria para resolver el problema.
  - Se tienden a demorar más tiempo en solucionar el problema.

UNIVERSIDAD DE SANTIAGO DE CHILE

Ejercicios

- Introducción
- Concepto Recursión
- Tipos de recursión
- Comparación con la iteración

Naruto es un personaje de ficción, su uso es solo para fines pedagógicos