Taller de Álgebra I

Clase 4 - Sumatorias

Implementación de sumatorias

¿Cómo podemos implementar la función *sumatoria* : $\mathbb{N} \rightarrow \mathbb{N}$, donde $sumatoria(n) = \sum_{i=1}^{n} i$

Para resolver este tipo de ejercicios, se puede pensar a las sumatorias como

$$sumatoria(n) = \sum_{i=1}^{n} i = n + \sum_{i=1}^{n-1} i \qquad \text{para n} > 1$$

Es decir:

$$sumatoria(n) = n + sumatoria(n-1) \qquad \text{para n} > 1$$

Ejercicios: otras sumatorias

Implementar y dar el tipo de las siguientes funciones simil Ejercicio 5 Práctica 2.

1. $f1(n) = \sum_{i=0}^{n} 2^i$, $n \in \mathbb{N}_0$.
2. $f2(n, q) = \sum_{i=1}^{n} q^i$, $n \in \mathbb{N}$ y $q \in \mathbb{R}$
3. $f3(n, q) = \sum_{i=1}^{2n} q^i$, $n \in \mathbb{N}_0$ y $q \in \mathbb{R}$
4. $f4(n, q) = \sum_{i=n}^{2n} q^i$, $n \in \mathbb{N}_0$ y $q \in \mathbb{R}$

Ejercicios

- Implementar una función `eAprox :: Integer -> Float` que aproxime el valor del número e a partir de la siguiente sumatoria:

$$\hat{e}(n) = \sum_{i=0}^{n} \frac{1}{i!}$$

- Definir la constante `e :: Float` como la aproximación de e a partir de los primeros 10 términos de la serie anterior.

Ejercicios

- Implementar una función `eAprox :: Integer -> Float` que aproxime el valor del número e a partir de la siguiente sumatoria:

$$\hat{e}(n) = \sum_{i=0}^{n} \frac{1}{i!}$$

- Definir la constante `e :: Float` como la aproximación de e a partir de los primeros 10 términos de la serie anterior.

¡Atención! A veces ciertas funciones esperan un `Float` y nosotros tenemos un `Int`. Para estos casos podemos utilizar la función `fromIntegral :: Int -> Float`.

Ejercicios (sumatorias dobles)

1. Implementar la siguiente función:

$$f(n, m) = \sum_{i=1}^{n} \sum_{j=1}^{m} i^j$$

2. Implementar una función `sumaPotencias q n m` que sume todas las potencias de la forma q^{a+b} con $1 \leq a \leq n$ y $1 \leq b \leq m$.

3. Implementar una función `sumaRacionales n m` que sume todos los números racionales de la forma p/q con $1 \leq p \leq n$ y $1 \leq q \leq m$.

Ejercicios de tarea

4. Implementar la siguiente función:

$$g_1(i, n) = \sum_{j=i}^{n} i^j$$

5. Implementar la siguiente función:

$$g_2(n) = \sum_{i=1}^{n} \sum_{j=i}^{n} i^j$$

6. Implementar la siguiente función:

$$g_3(n) = \sum_{\substack{i=1 \\ i \text{ es par}}}^{n} 2^i$$

7. Implementar una función que dado un n, sume todos los números naturales menores o iguales que n que tengan todos los dígitos iguales.