

Task 1

El programa se invoca de la siguiente forma

run a b c

Los parametros son:

- a : Indica si el sistema numerico será signado o no signado (0/1)
- b : Tamaño (en bits) de la parte entera del numero (≥ 0)
- c : Tamaño (en bits) de la parte decimal del numero (≥ 0)

Todos los inputs deben ser numeros. Además y c no pueden ser 0 a la vez Todas estas restricciones son validadas

El problema usa formulas logicas simples para resolver el problema en los casos signados y no signados La intuición detrás de las fórmulas se obtuvo observando la composición del número binario

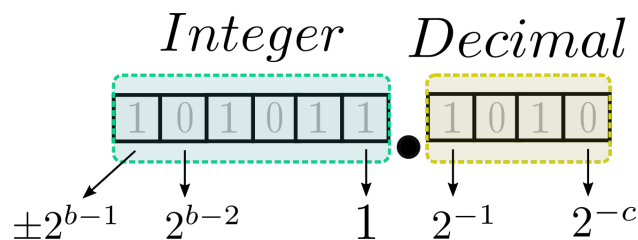


Figure 1: Visión del valor de cada bit

Como el bit menos significativo vale 2^{-c} luego la resolución es 2^{-c} y esto funciona tanto en el caso signado como en el no signado

Sin embargo, para computar el rango debemos pensar en los dos casos por separado, mirando cuál es el más grande número (que llamaremos r) y el más chico número (que llamaremos l) que podemos formar en cada caso En el caso no signado el número más chico es 0, y el más grande será

$$r = \underbrace{2^{-c} + 2^{-c+1} + \dots + 2^{-1}}_{1-2^{-c}} + \underbrace{1 + 2^1 + \dots + 2^{b-1}}_{2^b-1} = 2^b - 2^{-c}$$

Por lo tanto la resolución será $r - l = 2^b - 2^{-c}$

Para el caso signado notamos que el numero más chico es cuando solo el bit más significativo es 1

$$l = -2^{b-1}$$

Y el más grande es cuando todos los bits son 1 excepto el más significativo

$$r = 2^{-c} + 2^{-c+1} + \dots + 2^1 + 1 + 2^1 + \dots + 2^{b-2} = 2^{b-1} - 2^{-c}$$

Por lo tanto la resolución es $r - l = 2^{b-1} - 2^{-c} + 2^{b-1} = 2^b - 2^{-c}$ Que es la misma, lo cual es razonable ya que podemos expresar la misma cantidad de numeros con la misma cantidad de bits y esto no depende si el sistema es o no signado