

Clase #05 de 27

Proceso de Desarrollo de Funciones & Tipos Punto-Flotante

May 7, Miércoles

Agenda para esta clase

- Revisión Trabajo #0: “Hello, World” en C++
- Especificación de Operaciones de Tipos de Datos mediante Funciones como
- Proceso de Desarrollo de Funciones
- Ejercicio: DosVeces(n)
- Tipos Punto-Flotante

Especificación de Operaciones de Tipos de Datos mediante Funciones como

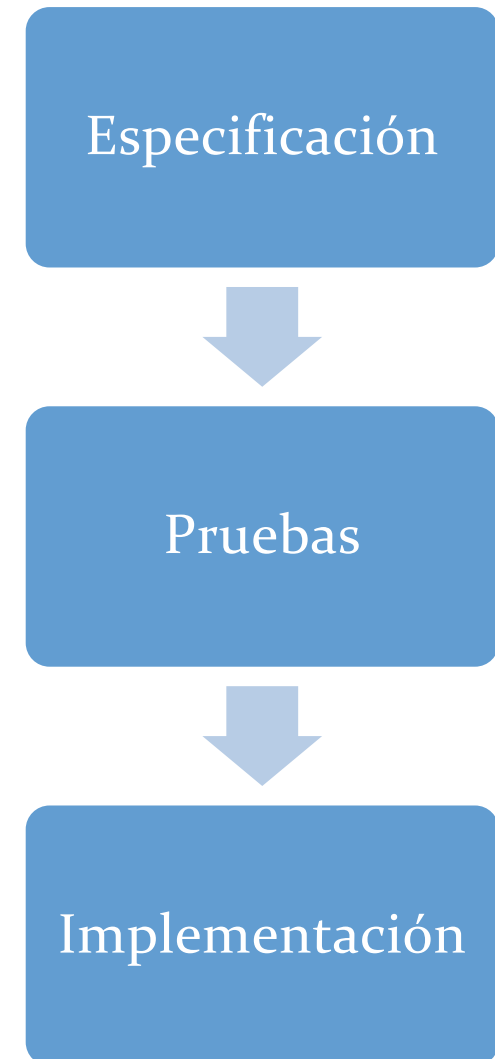
Especificación de Operaciones de Tipos de Datos mediante Funciones como

- Tipo **Coordenadas**
 - Distancia: $\text{Coord} \times \text{Coord} \rightarrow \mathbb{R}$
- Tipo **Racionales**
 - *Valores*
 - $Q = \{ q = (n,d) / n,d \in \mathbb{N}, d \neq 0 \}$
 - $Q = \{ q = (n,d) / n \in \mathbb{Z}, d \in \mathbb{N}, d \neq 0 \}$
 - $(1,3), (1,1), (0,7), (7,0), -(1,3)$
 - *Operaciones*
 - $\text{Suma} : Q \times Q \rightarrow Q / \text{Suma}(a, b) = c$
 - $c = (a_n \cdot b_d + b_n \cdot a_d , a_d \cdot b_d)$
- Tipo **Cadenas**
 - $\text{EsCapicúa} : \Sigma^* \rightarrow \mathbb{B}$
 - $\text{EsCapicúa}(aba) = V$

Proceso de Desarrollo de Funciones

Proceso de Desarrollo de Funciones

- Proceso General
 1. Definir una *especificación* matemática de una función.
 2. Definir pruebas en base a la especificación.
 3. Programar una *implementación* en base a la especificación.
- Proceso Detallado
 1. Especificación de la función.
 2. Diseño de Pruebas, con partición del conjunto de los datos de prueba
 3. Codificación de las pruebas.
 4. Compilación de las pruebas.
 5. Implementación de la función.
 6. Compilación de la implementación.
 7. Ejecución.
 8. Evaluación de resultados.



Ejercicio: Desarrollar Función

Twice(n)

Doble ó duplo de un número, es decir un número dos veces

Ejercicio: Especificar e Implementar Twice(n)

- Calcula el *doble* ó *duplo* de un número, es decir un número *dos veces*.
- Es posible expresarlo como producto entre dos y el parámetro o como la adición del parámetro con el mismo.

$$\text{Twice: } \mathbb{Z} \rightarrow \mathbb{Z} / \text{Twice}(n) = 2n$$

```
#include<cassert>
```

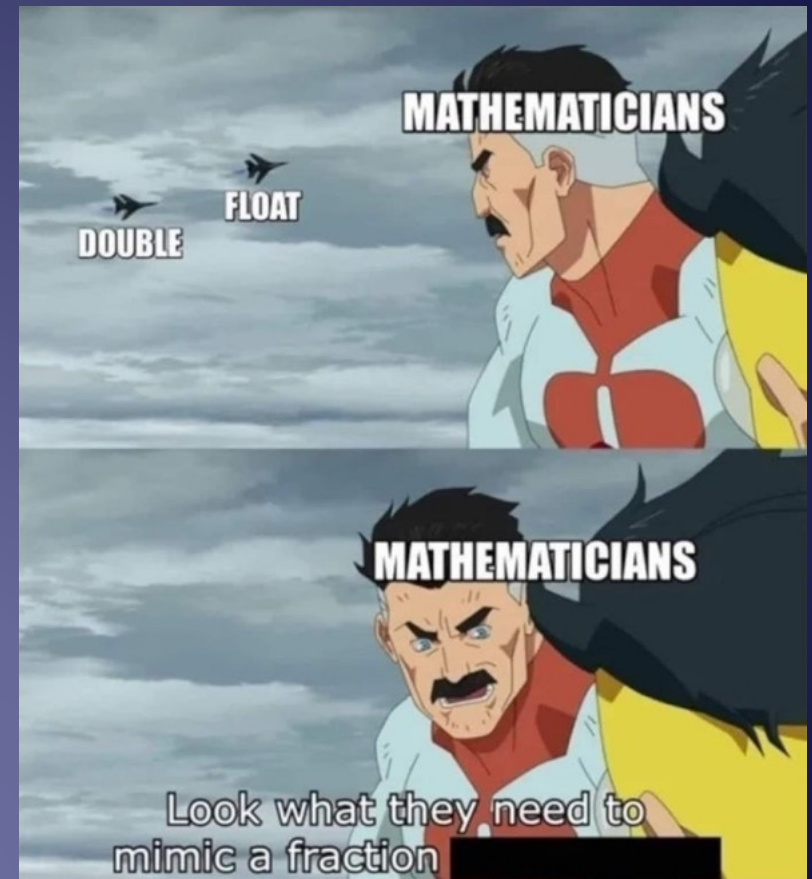
```
int Twice(int);
```

```
int main(){  
    assert( -14 == Twice(-7) );  
    assert(  0 == Twice(0) );  
    assert( 42 == Twice(21) );  
}
```

```
int Twice(int n){return 2*n;}
```


Tipos Punto-Flotante

Una aproximación a los números reales



Representación de Reales en tipo double

- Reales es un conjunto infinito
- Se representan electrónicamente con un conjunto finito de bits, generalmente 32 o 64 bits
- Entonces, la cantidad de valores que se pueden representar es finita
- La representación se define en IEEE 60559 y IEEE 754
- Sigue la notación científica, pero en base 2
- No todos los valores reales se pueden representar precisamente en base decimal, por ejemplo $1/3$ se se aproxima con 1.333
- De la misma forma no todos los valores reales se pueden representar de forma exacta en base binaria, por ejemplo $1/10$
- De la escala continua de reales, la representación IEEE puede representar solo algunos y esa distribución no es uniforme
- En C++ hay varios tipos que representan reales, nosotros usamos double, que es el tipo por defecto
- Otros lenguajes tienen tipos equivalentes
- Es un tema que se desarrolla en la asignatura “Arquitectura de Computadoras”
- Artículos sobre el tema: [Universidad de Texas](#), [Bit Bashing](#),

Recuerden repasar para qué se usan

Uso de los Tipos Numéricos:

Tipo Natural (unsigned) versus Tipo Flotante (double)

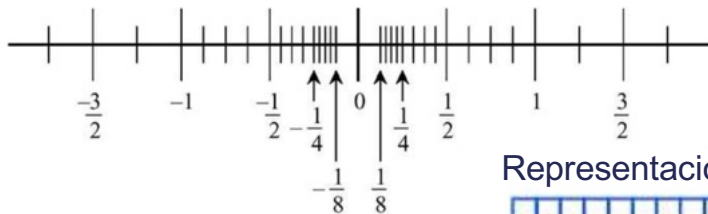
Contable

- En Matemática:
 - \mathbb{N} (Natural, infinitos)
- En C++:
 - unsigned (subrango **finito**)
 - Hay otros
- Cantidades
- Discreto
- Ejemplos
 - Stock
 - Participantes de una clase
 - Unidades monetarias (centavos)

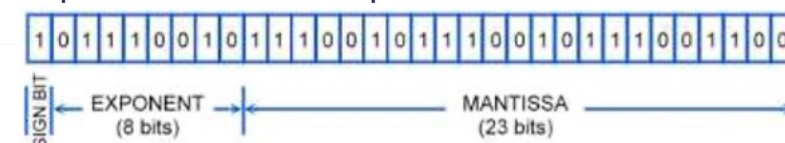
No contable

- En Matemática
 - \mathbb{R} (Real, infinitos)
- En C++:
 - Aproximación con double (subconjunto **finito** de \mathbb{R})
 - Hay otros
- Continuo
- Mediciones \Rightarrow error
- Ejemplos
 - Temperatura de una habitación
 - Volumen de agua de un recipiente
 - Probabilidad de ocurrencia

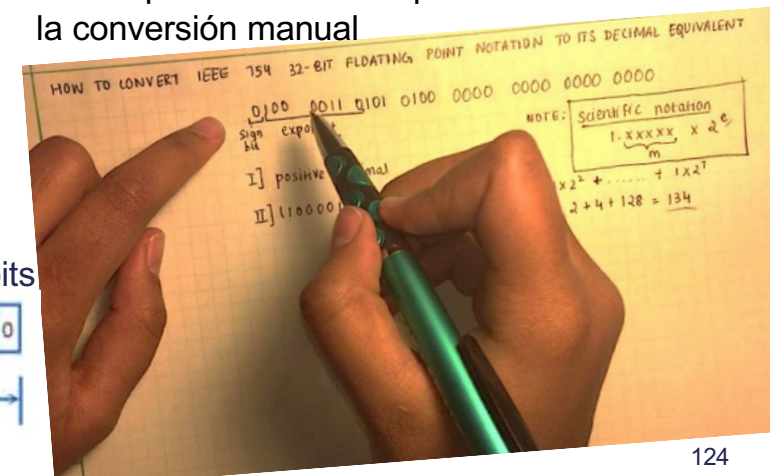
Distribución no uniforme



Representación en bloque con cantidad finita de bits



En “Arquitectura de Computadoras” realizan la conversión manual



Ejemplo de Representación de algunos Valores

= 0x0000000000000000

$$1 \times 2^{-1022} \times 0 = 0$$

= 0x3FF0000000000000

$$1 \times 2^0 \times 1 = 1$$

[illegible]

$$-1 \times 2^0 \times 1 = -1$$

[illegible]

$$1 \times 2^1 \times 1 = 2$$

= 0xC000000000000000

$$-1 \times 2^1 \times 1 = -2$$

[illegible]

$$1 \times 2^2 \times 1.25 = 5$$

[illegible]

$$1 \times 2^{-1} \times 1 = 0.5$$

[illegible]

$$1 \times 2^{-4} \times 1.6 = 0.1$$

[illegible]

$$1 \times 2^{-3} \times 1.6 = 0.2$$

[illegible]

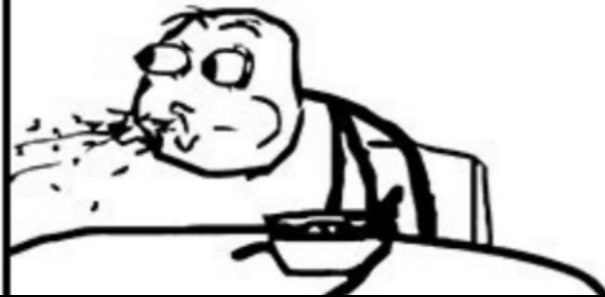
$$1 \times 2^{-2} \times 1.2 = 0.3$$

$0.1 + 0.2$



It will be 0.3

0.30000000000000004



Me after learning about
floating point inaccuracy



Términos de la clase #05

Definir cada término con la bibliografía

- Proceso de Desarrollo de Funciones
- Especificación
- Pruebas
- Implementación
- Tipos Punto-Flotante



Tareas para la próxima clase

1. Investigar sobre la comparación de tipos punto-flotante.

¿Consultas?

Fin de la clase