Representación de la información: Representación de números fraccionarios y reales

Representación de datos numéricos

- Fraccionarios:
  - Punto Fijo.
  - Punto Flotante.

Números fraccionarios

- Los números fraccionarios son aquellos racionales que no son enteros, y se escriben como una razón, fracción o cociente de dos enteros.
- Cuando se representan como decimales, la parte decimal puede ser finita o periódica (no finita, pero que se repite).

Números fraccionarios

- Los números fraccionarios son aquellos racionales que no son enteros, y se escriben como una razón, fracción o cociente de dos enteros.
- Cuando se representan como decimales, la parte decimal puede ser finita o periódica (no finita, pero que se repite).
- Por ejemplo $0,75$ también puede escribirse como $\frac{3}{4}$.
- Por ejemplo $0,\overline{3}$ también puede escribirse como $\frac{1}{3}$.

Números irracionales

- Por otro lado, existen números reales que no son racionales (no existe una fracción entre enteros que los represente) pero tienen una representación decimal con parte entera y una parte decimal no periódica e infinita.

Números irracionales

- Por otro lado, existen números reales que no son racionales (no existe una fracción entre enteros que los represente) pero tienen una representación decimal con parte entera y una parte decimal no periódica e infinita.
- Por ejemplo $\sqrt{2}$ no puede escribirse como una fracción de enteros, pero si como

Números irracionales

- Por otro lado, existen números reales que no son racionales (no existe una fracción entre enteros que los represente) pero tienen una representación decimal con parte entera y una parte decimal no periódica e infinita.
- Por ejemplo $\sqrt{2}$ no puede escribirse como una fracción de enteros, pero si como
  1,414213562373095048801688724209698078569671875376
  948073176679737990732478462107038850387534327641 57
  273501384623091229702492483605585073721264412149 70
  999358314132226659275055927557999505011527820605 71
  470109559971605970274534596862014728517418640889 19
  860955232923048430871432145083976260362799525140 79
  896872533965463318088296406206152583523950547457 50
  287759961729835575220337531857011354374603408498 84
  716038689997069900481503054402779031645424782306 84
  929369186215805784631115966687130130156185689872 37
  23528850926486124949771542183342042856860601468247

Expresión General Extendida

La coma (o punto decimal) señala el lugar donde los exponentes de la base de la expresión general se hacen negativos.

Sea un número $N_{(b)} = X_k X_{k-1} ... X_1 X_0, X_{-1} X_{-2} ... X_{-q}$, donde cada X_i son sus dígitos en base b. Su representación en decimal $W_{(10)}$:

$$
\begin{aligned}
N_{(b)} = & \sum_{i=-q}^{k} X_i \times b^i \\
= & X_k \times b^k + X_{k-1} \times b^{k-1} ... \\
& X_1 \times b^1 + X_0 \times b^0 + X_{-1} \times b^{-1} ... \\
& X_{-q+1} \times b^{-q+1} + X_{-q} \times b^{-q} \\
= & W_{(10)}
\end{aligned}
$$

Expresión General Extendida

Ejemplos:

- $3{,}14_{(10)} = 3\times10^0 + 1\times10^{-1} + 4\times10^{-2} = 3{,}14_{(10)}$
- $10{,}01_{(10)} = 1\times2^1 + 0\times2^0 + 0\times2^{-1} + 1\times2^{-2} = 2{,}25_{(10)}$

Conversión de decimal a binario

1. Se separan la parte entera y la parte fraccionaria.
2. Se convierte la parte entera a binario.
3. La parte fraccionaria se multiplica por 2 y se toma la parte entera del resultado. Este dígito binario se agrega al resultado final.
4. Se repite el paso 3 con la nueva parte fraccionaria obtenida, hasta que ésta sea 0, o hasta lograr la precisión deseada.
5. La parte entera de cada uno de los resultados del paso 4, forman los dígitos del número en la base en orden de izquierda a derecha.
6. Sumamos la parte entera a la parte fraccionaria.

Conversión de decimal a binario

Ejemplo: Convertir el número $3{,}625_{(10)}$ a base 2.

1. Separamos parte entera (3) y parte fraccionaria (0,625).
2. La parte entera se convierte a base 2 como entero sin signo (utilizando el método de la división) dando $11_{(2)}$.

Conversión de decimal a binario

Ejemplo: Convertir el número $3{,}625_{(10)}$ a base 2.

1. Separamos parte entera (3) y parte fraccionaria (0,625).
2. La parte entera se convierte a base 2 como entero sin signo (utilizando el método de la división) dando $11_{(2)}$.
3. Expresamos la parte fraccionaria:
    1. Multiplicamos la parte fraccionaria hasta obtener cero o una precisión de 5 dígitos:

Conversión de decimal a binario

Ejemplo: Convertir el número $3{,}625_{(10)}$ a base 2.

1. Separamos parte entera (3) y parte fraccionaria (0,625).
2. La parte entera se convierte a base 2 como entero sin signo (utilizando el método de la división) dando $11_{(2)}$.
3. Expresamos la parte fraccionaria:
    1. Multiplicamos la parte fraccionaria hasta obtener cero o una precisión de 5 dígitos: $0{,}625 \times 2 \quad = 1{,}25$

Conversión de decimal a binario

Ejemplo: Convertir el número $3{,}625_{(10)}$ a base 2.

1. Separamos parte entera (3) y parte fraccionaria (0,625).
2. La parte entera se convierte a base 2 como entero sin signo (utilizando el método de la división) dando $11_{(2)}$.
3. Expresamos la parte fraccionaria:
    1. Multiplicamos la parte fraccionaria hasta obtener cero o una precisión de 5 dígitos: $0{,}625 \times 2 = 1{,}25$
       $0{,}25 \times 2 = 0{,}5$

Conversión de decimal a binario

Ejemplo: Convertir el número $3{,}625_{(10)}$ a base 2.

1. Separamos parte entera (3) y parte fraccionaria (0,625).
2. La parte entera se convierte a base 2 como entero sin signo (utilizando el método de la división) dando $11_{(2)}$.
3. Expresamos la parte fraccionaria:
   1. Multiplicamos la parte fraccionaria hasta obtener cero o una precisión de 5 dígitos:
   $$\begin{aligned} 0{,}625 \times 2 &= 1{,}25 \\ 0{,}25 \times 2 &= 0{,}5 \\ 0{,}5 \times 2 &= 1{,}0 \end{aligned}$$

Conversión de decimal a binario

Ejemplo: Convertir el número $3{,}625_{(10)}$ a base 2.

1. Separamos parte entera (3) y parte fraccionaria (0,625).
2. La parte entera se convierte a base 2 como entero sin signo (utilizando el método de la división) dando $11_{(2)}$.
3. Expresamos la parte fraccionaria:
   1. Multiplicamos la parte fraccionaria hasta obtener cero o una precisión de 5 dígitos:
   $$\begin{aligned} 0{,}625 \times 2 &= 1{,}25 \\ 0{,}25 \times 2 &= 0{,}5 \\ 0{,}5 \times 2 &= 1{,}0 \end{aligned}$$
   2. La parte fraccionaria es 0,101.

Conversión de decimal a binario

Ejemplo: Convertir el número $3{,}625_{(10)}$ a base 2.

1. Separamos parte entera (3) y parte fraccionaria (0,625).
2. La parte entera se convierte a base 2 como entero sin signo (utilizando el método de la división) dando $11_{(2)}$.
3. Expresamos la parte fraccionaria:
   1. Multiplicamos la parte fraccionaria hasta obtener cero o una precisión de 5 dígitos: $$\begin{aligned} 0{,}625 \times 2 &= 1{,}25 \\ 0{,}25 \times 2 &= 0{,}5 \\ 0{,}5 \times 2 &= 1{,}0 \end{aligned}$$
   2. La parte fraccionaria es 0,101.
4. $3{,}625_{(10)} = 11_{(2)} + 0{,}101_{(2)} = 11{,}101_{(2)}$

Representación de Punto Fijo *(n,k)*

- Se define una cantidad fija $(n - k)$ de **bits** para la parte entera y k **bits** para la parte fraccionaria.
- La computadora puede tratar aritméticamente a todos los números como si fueran enteros.
- Conversión de decimal a Punto Fijo *(n,k)*:
  1. Obtener la representación binaria del valor absoluto del número, con una precisión de k.
  2. Expresar la parte entera y fraccionaria en la cantidad adecuada de **bits**. Si es necesario completar con ceros, a la izquierda para la parte entera, y a la derecha en la fraccionaria.
  3. Si el número es negativo, aplicar la operación de complemento a 2.

Conversión de decimal a Punto Fijo *(n,k)*

Ejemplo, representar $-7,33$ en Punto fijo *(8,4)*:

Conversión de decimal a Punto Fijo *(n,k)*

Ejemplo, representar $-7,33$ en Punto fijo *(8,4)*:

1. $|-7,33_{(10)}| \simeq 111{,}0101_{(2)}$, con una precisión de k.

Conversión de decimal a Punto Fijo *(n,k)*

Ejemplo, representar $-7,33$ en Punto fijo *(8,4)*:

1. $|-7,33_{(10)}| \simeq 111{,}0101_{(2)}$, con una precisión de k.
2. Expresamos el número en 4**b** para la para la parte entera, y 4**b** para la parte fraccionaria: 01110101

Conversión de decimal a Punto Fijo *(n,k)*

Ejemplo, representar $-7,33$ en Punto fijo *(8,4)*:

1. $|-7,33_{(10)}| \simeq 111{,}0101_{(2)}$, con una precisión de k.
2. Expresamos el número en 4**b** para la para la parte entera, y 4**b** para la parte fraccionaria: 01110101
3. Como el número es negativo, aplicamos la operación de *C2*: $C2(01110101) = 10001011 \simeq -7,33_{(10)}$

Representación de Punto Fijo

- Adecuada para problemas con datos de magnitudes y precisiones similares.
- Aplicada en sistemas de tiempo real.
- Sistemas donde es importante la velocidad de cómputo, como juegos.
- Sistemas empotrados o embebidos con procesadores de recursos limitados.

¿Qué pasa si no es suficiente la cantidad de bits que tenemos para la parte fraccionaria? ¿Es realmente $1000\,1011 = -7,33_{(10)}$ con *PF(8,4)*?

¿Qué pasa si no es suficiente la cantidad de bits que tenemos para la parte fraccionaria? ¿Es realmente $1000\,1011 = -7,33_{(10)}$ con *PF(8,4)*?

El número almacenado será una aproximación al número original.

¿Qué pasa si no es suficiente la cantidad de bits que tenemos para la parte fraccionaria? ¿Es realmente $1000\,1011 = -7,33_{(10)}$ con *PF(8,4)*?

El número almacenado será una aproximación al número original.

Si quisiéramos conocer de **cuánto** es ese error, deberíamos calcular la diferencia entre el número que queremos representar, y el número representado por la computadora.

Representación de fraccionarios

- ¿Qué pasa si tenemos magnitudes muy diferentes?

Representación de fraccionarios

- ¿Qué pasa si tenemos magnitudes muy diferentes?
- Por ejemplo, si se necesita calcular el tiempo en que la luz recorre una millonésima de milímetro, la fórmula relacionará la velocidad de la luz en $\frac{m}{s}$ (unos $300\,000\,000 \frac{m}{s}$) con el tamaño en metros de un nanómetro ($0{,}000000001m$).

Notación Científica

- Los números se expresan en una forma estandarizada que consiste de un coeficiente y una mantisa multiplicada por una potencia de 10: $m \times 10^x$

Notación Científica

- Los números se expresan en una forma estandarizada que consiste de un coeficiente y una mantisa multiplicada por una potencia de 10: $m \times 10^x$

$$
\begin{aligned}
d =& 1 \times 10^{-9} m \\
v =& 3 \times 10^{8} \frac{m}{s}
\end{aligned}
$$

Notación Científica

- Los números se expresan en una forma estandarizada que consiste de un coeficiente y una mantisa multiplicada por una potencia de 10: $m \times 10^x$

$$d = 1 \times 10^{-9} m$$

$$v = 3 \times 10^{8} \frac{m}{s}$$

$$t = \frac{d}{v}$$

Notación Científica

- Los números se expresan en una forma estandarizada que consiste de un coeficiente y una mantisa multiplicada por una potencia de 10: $m \times 10^x$

$$
\begin{aligned}
d =& 1 \times 10^{-9} m \\
v =& 3 \times 10^{8} \frac{m}{s}
\end{aligned}
$$

$$
t = \frac{d}{v} = \frac{1 \times 10^{-9} m}{3 \times 10^{8} \frac{m}{s}}
$$

Notación Científica

- Los números se expresan en una forma estandarizada que consiste de un coeficiente y una mantisa multiplicada por una potencia de 10: $m \times 10^x$

$$
\begin{aligned}
d =& 1 \times 10^{-9} m \\
v =& 3 \times 10^{8} \frac{m}{s}
\end{aligned}
$$

$$
\begin{aligned}
t =& \frac{d}{v} = \frac{1 \times 10^{-9} m}{3 \times 10^{8} \frac{m}{s}} \\
=& \left(\frac{1}{3} \times 10^{-9-8} s\right)
\end{aligned}
$$

Notación Científica

- Los números se expresan en una forma estandarizada que consiste de un coeficiente y una mantisa multiplicada por una potencia de 10: $m \times 10^x$

$$
\begin{aligned}
d =& 1 \times 10^{-9} m \\
v =& 3 \times 10^{8} \frac{m}{s}
\end{aligned}
$$

$$
\begin{aligned}
t =& \frac{d}{v} = \frac{1 \times 10^{-9} m}{3 \times 10^{8} \frac{m}{s}} \\
=& (\frac{1}{3} \times 10^{-9-8} s) \\
=& 0,\overline{3} \times 10^{-17} s
\end{aligned}
$$

Notación Científica Normalizada

- El coeficiente m debe ser un valor mayor o igual que 1 y menor que 10.

Notación Científica Normalizada

- El coeficiente m debe ser un valor mayor o igual que 1 y menor que 10.
- Ejemplo: $t = 0,\overline{3} \times 10^{-17} s$

Notación Científica Normalizada

- El coeficiente m debe ser un valor mayor o igual que 1 y menor que 10.
- Ejemplo: $t = 0,\overline{3} \times 10^{-17} s = 3,\overline{3} \times 10^{-18} s$

Formato que utiliza la notación científica normalizada:

- Se utiliza 1**b** para el signo.
- Se utilizan n**b** para el exponente en exceso a $2^{n-1} - 1$.
- Se utilizan k**b** para la mantisa (parte decimal del número).

Una versión didáctica del formato:

- Se utiliza 1**b** para el signo.
- Se utilizan 4**b** para el exponente en exceso a 7.
- Se utilizan 3**b** para la mantisa.

Conversión de decimal a punto flotante

1. Separar el signo y escribir el valor absoluto de n en base 2.
2. Escribir el resultado en notación científica normalizada.
3. Expresar el exponente obtenido en el paso anterior, en exceso.
4. El coeficiente calculado se guarda sin su parte entera en la parte de mantisa.

Conversión de decimal a punto flotante

Ejemplo: Expresar el decimal $-5,5$ en *MiniFloat*.

- Necesitamos averiguar s, e y m.

Conversión de decimal a punto flotante

Ejemplo: Expresar el decimal $-5,5$ en *MiniFloat*.

- Necesitamos averiguar s, e y m.
- n es negativo, luego $s = 1$.

Conversión de decimal a punto flotante

Ejemplo: Expresar el decimal $-5,5$ en *MiniFloat*.

- Necesitamos averiguar s, e y m.
- n es negativo, luego $s = 1$.
- El valor absoluto en binario es: $101,1$.

Conversión de decimal a punto flotante

Ejemplo: Expresar el decimal $-5,5$ en *MiniFloat*.

- Necesitamos averiguar s, e y m.
- n es negativo, luego $s = 1$.
- El valor absoluto en binario es: $101,1$.
- Normalizado: $1,011 \times 2^2$ por lo que la mantisa es $m = 011$.

Conversión de decimal a punto flotante

Ejemplo: Expresar el decimal $-5,5$ en *MiniFloat*.

- Necesitamos averiguar s, e y m.
- n es negativo, luego $s = 1$.
- El valor absoluto en binario es: $101,1$.
- Normalizado: $1,011 \times 2^2$ por lo que la mantisa es $m = 011$.
- El exponente se representa en exceso a 7:
  $e = 2 + 7 = 9 = 1001_2$

Conversión de decimal a punto flotante

Ejemplo: Expresar el decimal $-5,5$ en *MiniFloat*.

- Necesitamos averiguar s, e y m.
- n es negativo, luego $s = 1$.
- El valor absoluto en binario es: $101,1$.
- Normalizado: $1,011 \times 2^2$ por lo que la mantisa es $m = 011$.
- El exponente se representa en exceso a 7:
  $e = 2 + 7 = 9 = 1001_2$
- Resultado Final: $1100\,1011$. En hexadecimal **CB**.

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.
- Convertimos el hexadecimal a binario: $\texttt{5C}_{(16)} = 0101\,1100$.

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.
- Convertimos el hexadecimal a binario: $\texttt{5C}_{(16)} = 0101\,1100$.
- Recuperando los valores:$s = 0$, $e = 1011_{(2)} = 11_{(10)}$, $m = 0{,}100$.

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.
- Convertimos el hexadecimal a binario: $\texttt{5C}_{(16)} = 0101\,1100$.
- Recuperando los valores:$s = 0$, $e = 1011_{(2)} = 11_{(10)}$, $m = 0{,}100$.
- Remplazamos los valores en la siguiente formula:

$$N = (-1)^s \times (1 + m)_{(2)} \times 2^{(e-7)}$$

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.
- Convertimos el hexadecimal a binario: $\texttt{5C}_{(16)} = 0101\,1100$.
- Recuperando los valores:$s = 0$, $e = 1011_{(2)} = 11_{(10)}$, $m = 0{,}100$.
- Remplazamos los valores en la siguiente formula:

$$\begin{aligned} N &= (-1)^s \times (1 + m)_{(2)} \times 2^{(e-7)} \\ &= (-1)^0 \times (1 + 0{,}100)_{(2)} \times 2^{(11_{(10)}-7)} \end{aligned}$$

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.
- Convertimos el hexadecimal a binario: $\texttt{5C}_{(16)} = 0101\,1100$.
- Recuperando los valores:$s = 0$, $e = 1011_{(2)} = 11_{(10)}$, $m = 0{,}100$.
- Remplazamos los valores en la siguiente formula:

$$\begin{aligned} N &= (-1)^s \times (1+m)_{(2)} \times 2^{(e-7)} \\ &= (-1)^0 \times (1+0{,}100)_{(2)} \times 2^{(11_{(10)}-7)} \\ &= (-1)^0 \times (1{,}100)_{(2)} \times 2^{(4_{(10)})} \end{aligned}$$

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.
- Convertimos el hexadecimal a binario: $\texttt{5C}_{(16)} = 0101\,1100$.
- Recuperando los valores:$s = 0$, $e = 1011_{(2)} = 11_{(10)}$, $m = 0{,}100$.
- Remplazamos los valores en la siguiente formula:

$$
\begin{aligned}
N &= (-1)^s \times (1 + m)_{(2)} \times 2^{(e-7)} \\
&= (-1)^0 \times (1 + 0{,}100)_{(2)} \times 2^{(11_{(10)}-7)} \\
&= (-1)^0 \times (1{,}100)_{(2)} \times 2^{(4_{(10)})} \\
&= 1 \times (11000{,}0)_{(2)}
\end{aligned}
$$

Conversión de punto flotante a decimal

Ejemplo: ¿Qué número esta representado en el *MiniFloat* **5C**?

- Necesitamos averiguar s, e y m.
- Convertimos el hexadecimal a binario: $\texttt{5C}_{(16)} = 0101\,1100$.
- Recuperando los valores:$s = 0$, $e = 1011_{(2)} = 11_{(10)}$, $m = 0{,}100$.
- Remplazamos los valores en la siguiente formula:

$$
\begin{aligned}
N &= (-1)^s \times (1+m)_{(2)} \times 2^{(e-7)} \\
&= (-1)^0 \times (1+0{,}100)_{(2)} \times 2^{(11_{(10)}-7)} \\
&= (-1)^0 \times (1{,}100)_{(2)} \times 2^{(4_{(10)})} \\
&= 1 \times (11000{,}0)_{(2)} \\
&= 1 \times 24_{(10)} = 24_{(10)}
\end{aligned}
$$

- Precisión simple: 1**b** de signo, 8**b** para el exponente, y **23b** para la mantisa.
- Precisión doble: 1**b** de signo, 11**b** para el exponente, y **52b** para la mantisa.

- Precisión simple: 1**b** de signo, 8**b** para el exponente, y **23b** para la mantisa.
- Precisión doble: 1**b** de signo, 11**b** para el exponente, y 52**b** para la mantisa.

A pesar del incremento en la precisión, aun puede causar problemas: ¿Cuánto es $0{,}2 + 0{,}3$? (ver en la consola con `python3`)

- Fraccionarios:
  - Punto Fijo.
  - Punto Flotante.

¿Consultas?