Aritmética de los Números Enteros

Pablo L. De Nápoli

Departamento de Matemática
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Álgebra I - Segundo cuatrimestre de 2020

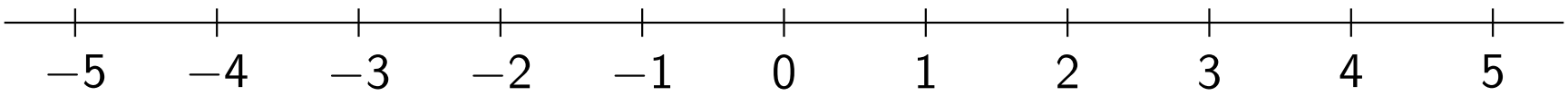
Parte I

Los números enteros

El conjunto de los Enteros

Notamos por $\mathbb{Z}$ al conjunto de los números enteros (positivos, negativos y cero)

$$\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\} = \mathbb{N} \cup \{0\} \cup \{-n : n \in \mathbb{N}\}$$



Si a es un entero su módulo o valor absoluto se define por

$$|a| = \begin{cases} a & \text{si} \quad a \geq 0 \\ -a & \text{si} \quad a < 0 \end{cases}$$

Notamos que si $a \in \mathbb{Z} \Rightarrow |a| \in \mathbb{N}_0$. Intuitivamente, el módulo es el número sin el signo. Ej:

$$|-3| = 3, |4| = 4, |0| = 0$$

También pueden pensarlo como la distancia de a al origen de coordenadas (0).

Operaciones con los enteros

En el conjunto $\mathbb{Z}$ de los números enteros, las operaciones de suma $a + b$, resta $a - b$ y producto $a \cdot b$ son siempre posibles (es un ejemplo de anillo).

Las reglas para operar con ellos las aprendieron en la escuela secundaria, por ejemplo:

Regla de los signos

Para todo $a, b \in \mathbb{Z}$,

$$\begin{aligned} a \cdot (-b) &= -(a \cdot b) \\ (-a) \cdot b &= -(a \cdot b) \\ (-a) \cdot (-b) &= a \cdot b \end{aligned}$$

Por ejemplo,

$$(-1) \cdot (-1) = 1$$

¿Pero se preguntaron alguna vez de donde viene esta regla? ...

Porqué es necesaria esta regla

Cuando un concepto matemático se generaliza, se procura preservar las propiedades formales que tenía el concepto que estamos generalizando. Por ejemplo, en los números naturales vale la ley distributiva:

$$(a+b)\cdot c = a\cdot c + b\cdot c$$

Al definir las operaciones con los números negativos, queremos que esta propiedad se preserve. Entonces tomando $a = 1, b = -1, -1$ nos queda

$$(1+(-1))\cdot(-1) = 1\cdot(-1) + (-1)\cdot(-1)$$

Pero $1+(-1) = 0$ por definición de inverso aditivo, y queremos que

$$0\cdot(-1) = 0, 1\cdot(-1) = -1$$

para preservar las propiedades

$$0\cdot a = 0, \quad 1\cdot a = a \quad \text{que valían en los naturales.}$$

Nos queda

$$0 = (-1) + (-1)(-1) \Rightarrow (-1)(-1) = 1$$

Parte II

Divisibilidad

En cambio, la división $a : b$, o lo que es equivalente, resolver la ecuación $b \cdot x = a$, no es siempre posible en los enteros.

Por ejemplo, la división $6 : 3$ es posible, ya que existe un entero 2 tal que $6 = 3 \times 2$. Mientras que la división $7 : 3$ no es posible, ya que no existe ningún entero c tal que $7 = 3c$. Esto motiva la siguiente definición:

Definición

Sean $a, b \in \mathbb{Z}$ dos enteros, $b \neq 0$, diremos que b divide a a, o que a es divisible por b, o que a es un múltiplo de b, o que b es un factor de a si existe algún entero $c \in \mathbb{Z}$ (necesariamente único) tal que $a = bc$. (De modo que $a : b = c$). Simbolizamos este hecho mediante la notación: $b|a$

Por ejemplo, 3 divide a 6, pero 3 no divide a 7.

Algunas Propiedades elementales de la divisibilidad

- Para cualquier $a \in \mathbb{Z}$, $a \neq 0$; $a|a$ (ya que $a = a \cdot 1$). Es decir, la relación de divisibilidad es reflexiva.
- Para cualquier $a \in \mathbb{Z}$, $a \neq 0$; tenemos que $a|0$.
- Para cualquier $a \in \mathbb{Z}$, $1|a$ y $-1|a$ (ya que $a = 1 \cdot a = (-1) \cdot (-a)$). Vemos que 1 y -1 son divisores universales.
  En la terminología usual en álgebra, esto se expresa diciendo que ± 1 son las unidades de $\mathbb{Z}$.
- Si $a|b$, entonces $(-a)|b$, $a|(-b)$ y $(-a)|(-b)$. Vale decir que para las cuestiones de divisibildad los números a y $-a$ son completamente equivalentes. (podemos ignorar el signo cuando estamos estudiando la divisibilidad entre dos números)

Transitividad

Proposición

Si $a|b$ y $b|c$ (siendo $a, b \neq 0$) entonces $a|c$, vale decir que la divisibilidad es una relación transitiva.

Demostración.

Como $a|b$, existe un entero e tal que

$$b = ae$$

y como $b|c$, existe un entero f tal que

$$c = bf$$

Sustituyendo en esta ecuación el valor de b, y usando la propiedad asociativa del producto:

$$c = (ae)f = a(ef)$$

Concluimos que $a|c$. □

Divisibilidad y Orden

Proposición

Si $b|a$ siendo $a, b \neq 0$, entonces $|b| \leq |a|$.

Demostración.

Si $b|a$, entonces existe un c tal que $a = bc$. Tomando módulo, tenemos que:

$$|a| = |b||c|$$

y como $|c| \geq 1$ (ya que si $c = 0$ entonces sería $a = 0$), concluimos que:

$$|a| \geq |b|$$

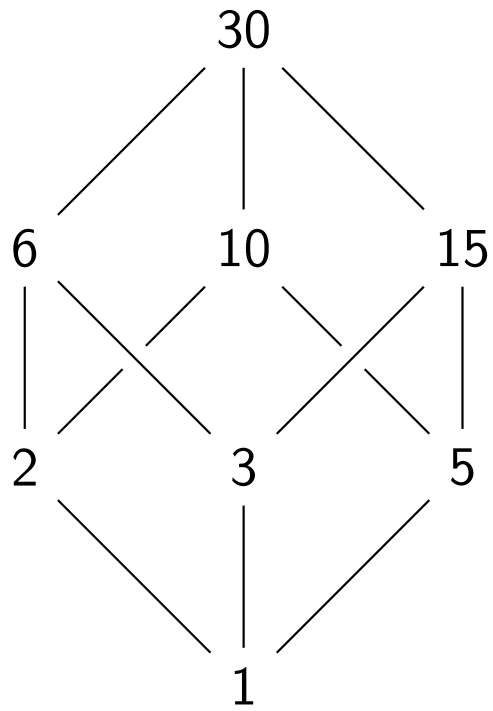
□

Corolario

Si $a|b$ y $b|a$ (siendo $a, b \neq 0$) entonces $a = \pm b$.

La divisibilidad como relación de orden en $\mathbb{N}$

Se deduce de lo anterior, que en $\mathbb{N}$ la divisibidad es una relación de orden (parcial) al ser reflexiva, antisimétrica y transitiva.
Podemos visualizar esto en el diagramma de Hasse de la divisibilidad entre los divisores de 30:



Notemos que no es un orden total, pues existen elementos incomparables. Por ejemplo 2 no divide a 3 ni 3 divide a 2.

Enteros asociados

En cambio, en el conjunto de enteros no nulos $\mathbb{Z} - \{0\}$ la divisibilidad no es antisimétrica pues

$$3|-3 \wedge -3|3 \wedge 3 \neq -1$$

y en consecuencia no es una relación de orden. Pero deducimos de lo anterior, que si en $\mathbb{Z} - \{0\}$ definimos la relación ser enteros asociados (o equivalentes para la divisibilidad) por

$$a \sim b \Leftrightarrow a|b \wedge b|a$$

esto define una relación de equivalencia y tenemos que

$$a \sim b \Leftrightarrow |a| = |b|$$

Para la relación de ser asociados la clase de equivalencia de a es $\{a, -a\}$.

La divisibilidad y las operaciones de suma y resta

Proposición

Si $a|b$ y $a|c$, se tiene que $a|b+c$ y que $a|b-c$.

Demostración.

Como $a|b$, existirá un entero e tal que:

$$b = ae$$

y como $a|c$, existirá otro entero f tal que:

$$c = af$$

Sumando estas dos ecuaciones, y aplicando la propiedad distributiva, tenemos que:

$$b + c = ae + af = a(e + f)$$

Concluimos que $a|b+c$. Similarmente, tenemos que:

$$b - c = ae - af = a(e - f)$$

Parte III

El algoritmo de división

El algoritmo de división en $\mathbb{N}$

Toda la aritmética gira en torno del siguiente hecho fundamental, conocido desde la escuela primaria: Si bien la división $a : b$ puede resultar imposible dentro de los enteros, siempre es posible efectuar una división aproximada, obteniendo un resto menor que el divisor. Por ejemplo: la división entera de 7 por 3 da un cociente de 2 con un resto de 1, y se tiene que $7 = 3 \times 2 + 1$.

El siguiente teorema, formaliza este hecho:

Teorema

Dados números naturales $a \in \mathbb{N}_0$ y $b \in \mathbb{N}$ existen únicos números naturales $a, b \in \mathbb{N}_0$ tales que $a = bq + r$ y $0 \leq r < b$.

Definición

En la situación del teorema anterior, diremos que q es el cociente y r el resto en la división entera de a por b.

Bonus Track: ¡En la computadora! (en Python 3)

división en Python 3

```
>>> 7/3
2.3333333333333335
>>> 7//3
        2
>>> 7%3
        1
>>> divmod(7,3)
        (2, 1)
```

Relación con la división en los reales

Si $x \in \mathbb{R}$, se define su parte entera

$$[x] = \text{máx}\{n \in \mathbb{Z} : n \leq x\}$$

Ejemplos:

$$[2{,}1] = 2, \quad [-2{,}1] = -3$$

Esta definición la usaremos hoy sólo con $x \geq 0$ (el caso más fácil de entender).
Observamos que si $a \in \mathbb{N}_0, b \in \mathbb{N}$,

$$q(a,b) = \left[\frac{a}{b}\right]$$

ya que

$$a = bq + r \Leftrightarrow \frac{a}{b} = q + d \text{ donde } d = \frac{r}{b}$$

y $0 \leq d < 1$. (d es la parte decimal de a/b)
Entonces

$$q(a,b) \leq \frac{a}{b} < q(a,b) + 1$$

y por lo tanto

$$q(a,b) = \left[\frac{a}{b}\right]$$

Idea de la demostración del teorema

Aunque en la escuela aprendieron un algoritmo para efectuar la división entera a partir de la representación decimal de los naturales, es interesante saber que el teorema puede demostrarse usando sólo las propiedades básicas de los naturales (por ejemplo los axiomas de Peano) sin hacer referencia a la representación decimal.

La idea para demostrarlo es que podemos efectuar la división por restas sucesivas, hasta obtener un resto menor que el divisor, por ejemplo para efectuar la división entera de 17 por 3

$$17 - 3 = 4, 14 - 3 = 11, 11 - 3 = 8, 8 - 3 = 5, 5 - 3 = 2$$

Hicimos 5 restas. Luego el cociente es 5. Y efectivamente $5 \cdot 3 + 2 = 17$

En mi apunte de números enteros, pueden ver una formalización tradicional de esto usando el principio del mínimo entero. Sin embargo, con esta formalización quizás no es claro porqué decimos que es un algoritmo.

Idea de la demostración (2)

En la clase de hoy intentaré presentarles un enfoque algorítmico a la misma demostración, usando algunas ideas clave de la computación científica. Estas ideas nos serán útiles cuando expliquemos otros algoritmos como el algoritmo de Euclides. Los que cursen computación las verán más en detalle en Algoritmos y Estructuras de datos I.

La idea básica de la demostración es que podemos formalizar el procedimiento de restas sucesivas definiendo recursivamente una función $division : \mathbb{N}_0 \times \mathbb{N}_0 \to \mathbb{N}_0$ $\text{division}(a, b) = (q(a, b), r(a, b))$ por

$$division(a, b) = \begin{cases} (0, a) & \text{si} \quad b < a \\ (q(a - b, b) + 1, r(a - b, b)) & \text{si} \quad b \geq a \end{cases}$$

Sin embargo, esta definición es más compleja que las definiciones recursivas que trabajamos hasta el momento. ¿Porqué funciona? ¡Veámosla en acción!

Bonus Track:¡En la computadora! (en Python 3)

Bonus track: Division Entera por restas sucesivas en Python 3

```
def division(a,b):
        print("division(",a,",",b,")","=",a-b)
        if a<b:
                nuevo_q =0
                nuevo_r =a
        else:
                nuevo_a=a-b
                print(a,"-",b,"=",nuevo_a)
                q,r= division(nuevo_a,b)
                nuevo_q = q+1
                nuevo_r = r
        print("division(",a,",",b,")=(",
                  nuevo_q,",",nuevo_r,")")
        return (nuevo_q,nuevo_r)
```

Bonus Track: Salida del programa

Division Entera de 17 por 3 por restas sucesivas en Python 3

```
division( 17 , 3 )
17 - 3 = 14
division( 14 , 3 )
14 - 3 = 11
division( 11 , 3 )
11 - 3 = 8
division( 8 , 3 )
8 - 3 = 5
division( 5 , 3 )
5 - 3 = 2
division( 2 , 3 )
division( 2 , 3 )=( 0 , 2 )
division( 5 , 3 )=( 1 , 2 )
division( 8 , 3 )=( 2 , 2 )
division( 11 , 3 )=( 3 , 2 )
division( 14 , 3 )=( 4 , 2 )
division( 17 , 3 )=( 5 , 2 )
```

Dos preguntas fundamentales

Pero ¿porqué funciona este enfoque?. ¿Porqué esta función queda bien definida? Tenemos que contestar dos preguntas:

- ¿Cómo podemos asegurar que este algoritmo siempre termina?
- ¿Cómo sabemos que va a devolver el resultado correcto?

¿Porqué termina?

La primer pregunta es fácil de responder: observemos que los números obteniendo por las restas sucesivas son cada vez más pequeños.
Por ejemplo, en el ejemplo anterior:

$$17 > 14 > 11 > 8 > 5 > \dots$$

Por el principio del mínimo entero, esta sucesión decreciente de enteros no negativos no puede ser infinita.
Tarde o temprano vamos a obtener un número menor que el divisor, y ahí el algoritmo se detiene.
En nuestra implementación informática, estos números son los que aparecen como primer argumento en cada llamada recursiva a nuestra función.

¿Cómo sabemos que el algoritmo da el resultado correcto?

Para formalizar esto observamos que hay una propiedad llamada invariante del algoritmo que se preserva en todos los pasos del algoritmo.

En este caso el invariante es justamente la condición que aparece en el enunciado del teorema

Invariante del algoritmo de división

$$a = b \cdot q(a, b) + r(a, b)$$

Comprobación de que se cumple el invariante

Es por inducción global en a.

- Si $a < b$ [y en particular si $a = 0$], el nuevo q es 0 y el nuevo r es a y $a = b \cdot 0 + a$.
- Si $a \geq b$, el nuevo q es $q(a, b) + 1$ y el nuevo r sigue siendo r. Ahora el algoritmo divide a $a - b$ por b y como $\tilde{a} = a - b < a$ (pues $b \geq 1$) podemos usar la hipótesis inductiva para decir que

$$a - b = b \cdot q + r \; q = q(a - b, b)$$

Pero sumando b a dos lados de la igualdad

$$a = b \cdot (q + 1) + r$$

El nuevo $\tilde{q} = q(a, b)$ lo calculamos por $\tilde{q} = q + 1$ y el nuevo $\tilde{r} = r(a, b)$ por $\tilde{r} = r$. Esto justamente hace que se siga cumpliendo:

$$a = b \cdot \tilde{q} + \tilde{r}$$

Bons track: chequeamos el invariante

Bonus track: Division Entera por restas con chequeo del invariante

```
def division(a,b):
        print("division(",a,",",b,")")
        if a<b:
                nuevo_q =0
                nuevo_r =a
        else:
                nuevo_a=a-b
                print(a,"-",b,"=",nuevo_a)
                q,r= division(nuevo_a,b)
                nuevo_q = q+1
                nuevo_r = r
        print("Invariante:",a,"=",b,"*",nuevo_q,
                "+",nuevo_r)
        invariante= (a==b*nuevo_q+nuevo_r)
        if not(invariante):
                print("Error")
        return (nuevo_q,nuevo_r)
```

Salida del programa

Division Entera de 17 por 3 por restas sucesivas en Python 3

```
division( 17 , 3 )
17 - 3
division( 14 , 3 )
14 - 3
division( 11 , 3 )
11 - 3
division( 8 , 3 )
8 - 3
division( 5 , 3 )
5 - 3
division( 2 , 3 )
Invariante: 2 = 3 * 0 + 2
Invariante: 5 = 3 * 1 + 2
Invariante: 8 = 3 * 2 + 2
Invariante: 11 = 3 * 3 + 2
Invariante: 14 = 3 * 4 + 2
Invariante: 17 = 3 * 5 + 2
```

Unicidad del cociente y resto

Supongamos que tenemos dos escrituras:

$$a = b \cdot q + r \; 0 \leq r < b$$

y

$$a = b \cdot \tilde{q} + \tilde{r} \; 0 \leq r < b$$

Podemos suponer $r \geq \tilde{r}$ (sino intercambiamos los nombres). Entonces

$$0 \leq r - \tilde{r} < b$$

Pero restando tenemos

$$b \cdot (q - \tilde{q}) = r - \tilde{r}$$

luego

$$b | r - \tilde{r}$$

Si fuera, $r \neq \tilde{r}$ podríamos deducir

$$b \leq r - \tilde{r}$$

¡absurdo!. Luego debe ser $r = \tilde{r}$ y volviendo a las identidades de arriba, deducimos que $q = \tilde{q}$.

¿Y si queremos trabajar con enteros que pueden ser negativos?

Teorema

Algoritmo de división en $\mathbb{Z}$ Dados números enteros $a, b \in \mathbb{Z}$ con $b \neq 0$ existen únicos números enteros tales que $a = bq + r$ y $0 \leq r < |b|$.

Pueden verlo en la sección 4.3 en el apunte de la profesora Krick. Dicho apunte usa la notación $r(a, b) = r_a(b)$. También emplea el enfoque algorítmico.