

tarea01

jfdelosrios

2022-06-27

Pregunta 1

¿Son los números enteros, \mathbb{Z} , un cuerpo? ¿Por qué? Razona tu respuesta

Pregunta 2

Realiza las siguientes sumas a mano y comprueba tu respuesta en R, Python u Octave:

- $(2 + 3i) + (1 + i)$

A mano

En R

$$3 + 4i$$

En Python

```
(2 + 3j) + (1 + 1j)
```

$(3+4j)$

En Octave

- $(1 + i) + (1 - i)$

A mano

En R

$$2 + 0i$$

En Python

```
(1 + 1j) + (1 - 1j)
```

$(2+0j)$

En Octave

- $(x^2 + x + 1) + (x - 1)$

A mano

En R

```
x = polynomial(coef = c(1, 1, 1)) + polynomial(coef = c(-1, 1))
x = toString(x)
result = x
#result = "(x^2 + x + 1) + (x - 1)"
result = result %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

$$x^2 + 2x$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'(x^2 + x + 1) + (x - 1)')
print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

$$x^2 + 2x$$

En Octave

Pregunta 3

Realiza los siguientes productos a mano y comprueba tu respuesta en R, Python u Octave:

- $(2 + 3i)(1 + i)$

A mano

En R

$$-1 + 5i$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'(2 + 3I)(1 + I)')
print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

$$-1 + 5i$$

En Octave

- $(1 + i)(1 - i)$

A mano

En R

$$2 + 0i$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'(1 + I)(1 - I)')
print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

$$2$$

En Octave

- $(x^2 + x + 1)(x - 1)$

A mano

En R

```
x = polynomial(coef = c(1, 1, 1)) * polynomial(coef = c(-1, 1))
x = toString(x)
result = x
#result = "(x^2 + x + 1) + (x - 1)"
result = result %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

$$x^3 - 1$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'(x^2 + x + 1)(x - 1)')
print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

$$x^3 - 1$$

En Octave

- $(x + 1)^2$

A mano

En R

```
x = polynomial(coef = c(1, 1))^2
x = toString(x)
result = x %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

$$x^2 + 2x + 1$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'(x + 1)^2')
print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

$$x^2 + 2x + 1$$

En Octave

- $(x + 1)(x - 1)$

A mano

En R

```
x = polynomial(coef = c(1, 1)) * polynomial(coef = c(-1, 1))
x = toString(x)
result = x %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

$$x^2 - 1$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'(x + 1)(x - 1)')
print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

$$x^2 - 1$$

En Octave

Pregunta 4

Calcula el módulo de los siguientes números complejos (realizando primero las operaciones pertinentes):

- $2 + 3i$

A mano

En R

```
x = "2 + 3*I"

result = x %>% y_fn("Abs") %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

$$\sqrt{13}$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'2 + 3I')
print('$$' + printing.latex(abs(p)) + '$$')
```

$$\sqrt{13}$$

En Octave

- i

A mano

En R

```
x = "I"

result = x %>% y_fn("Abs") %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

$$1$$

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy(r'I')
print('$$' + printing.latex(abs(p)) + '$$')
```

$$1$$

En Octave

- $(2 + 3i) + (1 + i)$

A mano

En R

```
x = "(2 + 3*I) + (1 + I)"  
  
result = x %>% y_fn("Abs") %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

5

En Python

```
from sympy import *  
from latex2sympy2 import latex2sympy  
  
p = latex2sympy('(2 + 3I) + (1 + I)')  
print('$$' + printing.latex(abs(p)) + '$$')
```

5

En Octave

- $(1 + i) + (1 - i)$

A mano

En R

```
x = "(1 + I) + (1 - I)"  
  
result = x %>% y_fn("Abs") %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

2

En Python

```
from sympy import *  
from latex2sympy2 import latex2sympy  
  
p = latex2sympy('(1 + I) + (1 - I)')  
print('$$' + printing.latex(abs(p)) + '$$')
```

2

En Octave

- $(2 + 3i)(1 + i)$

6

A mano

En R

```
x = "(2 + 3*I)*(1 + I)"  
  
result = x %>% y_fn("Abs") %>% y_fn("Simplify") %>% y_fn("TeXForm") %>% yac_str()
```

$$\sqrt{26}$$

En Python

```
from sympy import *  
from latex2sympy2 import latex2sympy  
  
p = latex2sympy('(2 + 3I)(1 + I)')  
print('$$' + printing.latex(abs(p)) + '$$')
```

$$\sqrt{26}$$

En Octave

Pregunta 5

Indica el grado de los siguientes polinomios (realizando primero las operaciones pertinentes):

- $2x + 2 \rightarrow$ Primer grado.
- $x^5 + 3x + 2 \rightarrow$ quinto grado.
- $(x^2 + x + 1)(x - 1) \rightarrow$ Tercer grado.
- $(x + 1)^2 \rightarrow$ Segundo grado.
- $(x + 1)(x - 1) \rightarrow$ Segundo grado.

Pregunta 6

¿Son iguales los siguientes polinomios?

- $(x + 1)^2$ y $x^2 + 1$

A mano

En R

```
polynomial(coef = c(1, 1))^2 == polynomial(coef = c(1, 0, 1))  
  
## [1] FALSE
```

En Python

```
import numpy  
  
numpy.poly1d([1, 1])**2 == numpy.poly1d([1, 0, 1])  
  
## False
```

En Octave

- $(x+1)^2 y x^2 + 2x + 1$

A mano

En R

```
polynomial(coef = c(1, 1))^2 == polynomial(coef = c(1, 2, 1))
```

```
## [1] TRUE
```

En Python

```
import numpy  
numpy.poly1d([1, 1])**2 == numpy.poly1d([1, 2, 1])
```

```
## True
```

En Octave

- $(x+1)^3 y x^3 + 1$

A mano

En R

```
polynomial(coef = c(1, 1))^3 == polynomial(coef = c(1, 0, 0, 1))
```

```
## [1] FALSE
```

En Python

```
import numpy  
numpy.poly1d([1, 1])**3 == numpy.poly1d([1, 0, 0, 1])
```

```
## False
```

En Octave

- $(x+1)^3 y x^3 + 3x^2 + 3x + 1$

A mano

En R

```
polynomial(coef = c(1, 1))^3 == polynomial(coef = c(1, 3, 3, 1))
```

```
## [1] TRUE
```

En Python

```
import numpy  
numpy.poly1d([1, 1])**3 == numpy.poly1d([1, 3, 3, 1])
```



```
## True
```

En Octave

- $(x+1)(x-1)$ y x^2-1

A mano

En R

```
polynomial(coef = c(1, 1)) * polynomial(coef = c(-1, 1)) == polynomial(coef = c(1, 0, 1))
```

```
## [1] FALSE
```

En Python

```
import numpy

numpy.poly1d([1, 1]) * numpy.poly1d([1, -1]) == numpy.poly1d([1, 0, 1])
```

```
## False
```

En Octave

- $(x-1)^2$ y x^2-2x+1

A mano

En R

```
polynomial(coef = c(-1, 1))^2 == polynomial(coef = c(1, -3, 1))
```

```
## [1] FALSE
```

En Python

```
import numpy

numpy.poly1d([1, -1])**2 == numpy.poly1d([1, -3, 1])
```

```
## False
```

En Octave

Pregunta 7

Encuentra las raíces de los siguientes polinomios:

- $2x+2$

A mano

En R

```
polyroot(c(2,2))
```

```
## [1] -1+0i
```

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy('2x + 2')

p = list(roots(p).keys())

print(p)

#print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

[-1]

En Octave

- $x^5 + 3x + 2$

A mano

En R

```
polyroot(c(2,3,0,0,1))

## [1] -0.8105357+0.0000000i -1.0000000-0.0000000i  0.9052679-1.283742i
## [4]  0.9052679+1.283742i
```

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy('x^5 + 3x + 2')

p=list(roots(p).keys())

print(p)

#print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

[]

En Octave

- $(x^2 + x + 1)(x - 1)$

A mano

En R

```
p1 = polynomial(coef = c(1, 1, 1)) * polynomial(coef = c(-1, 1))

polyroot(coef(p1))

## [1]  1.0-0.0000000i -0.5+0.8660254i -0.5-0.8660254i
```

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy('(x^2 + x + 1)(x - 1)')

p=list(roots(p).keys())

print(p)

#print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

[1, -1/2 - sqrt(3)I/2, -1/2 + sqrt(3)I/2]

En Octave

- $(x + 1)^2$

A mano

En R

```
p1 = polynomial(coef = c(1, 1, 1))^2
polyroot(coef(p1))

## [1] -0.5+0.8660254i -0.5+0.8660254i -0.5-0.8660254i -0.5-0.8660254i
```

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy('(x + 1)^2')

p=list(roots(p).keys())

print(p)

#print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

[-1]

En Octave

- $(x + 1)(x - 1)$

A mano

En R

```
p1 = polynomial(coef = c(1, 1)) * polynomial(coef = c(-1, 1))
polyroot(coef(p1))
```

```
## [1] 1+0i -1+0i
```

En Python

```
from sympy import *
from latex2sympy2 import latex2sympy

p = latex2sympy('(x + 1)(x - 1)')

p=list(roots(p).keys())

print(p)

#print('$$' + printing.latex(expand(simplify(p))) + '$$')
```

```
[-1, 1]
```

En Octave