

Python e Geogebra

Manuel Díaz Regueiro (Igaciencia)

Estes programas de software libre que nos ocupan teñen como característica común a de pretender ser amigables, fáciles e para todos, tal como transmiten a través dos respectivos lemas: *Computer Programming for Everybody*, Python, e *Dynamic Mathematics for Everyone*, Geogebra. Entre o dilema de facer ciencia para persoas escollidas e expertas e facer ciencia para todos, escollen a última opción.

Por outra banda, Python estase a converter nunha linguaxe de programación central para moitos proxectos máis amplos (Sage, Blender, Maya, Rhino...), ultimamente tamén para Geogebra, ao que lle aporta a potencia da súa linguaxe. Así, desta maneira, Geogebra deixa de ser un xoguete educativo para converterse nunha aplicación que ten multiplicada a súa utilidade.

Pero por que Python?

- A súa presenza e aplicación en numerosos lugares profesionais (na web, na bioinformática, administración de Unix e Linux, de Youtube, 3D, etc). Isto pode constatarase nunha simple busca que nos devolverá gran cantidade de bibliografía e libros sobre eses temas.

- Por ser Python a linguaxe libre de programación que usan os científicos. Polo que, naturalmente, hai moitas empresas que programan case exclusivamente en Python: Google, Youtube, a NASA e moitas outras menos coñecidas.

Se nos centramos nas posibilidades no mundo educativo:

- É un programa de software libre. Richard Stallman ilústranos con varias razóns da importancia do software libre na educación <http://www.gnu.org/education/edu-schools.es.html>

- Permite aprender a programar na secundaria. Guido van Rossum, o creador de Python, defende a liña de *Computer Programming For Everybody*, Programación de ordenador para todos, e propón Python como primeira linguaxe de programación na secundaria.

- Utilizando o seu Logo, permite rescatar as fantásticas aplicacións educativas desta linguaxe que poden volver á aula da man de Geogebra.

- Python é unha linguaxe apropiada para as Ciencias, Matemáticas, e Tecnoloxía -ao igual que Geogebra-, xunto con Geogebra súmanse as súas sinerxias e poténcianse ambos os dous programas. Véxase, por exemplo, a utilización con ExpEyes.

O profesor da área científica que non saiba Python e Geogebra ensinará unha Ciencia que dificilmente sintonizará co século XXI.

Exemplos sinxelos de uso da ventá Python de Geogebra

Iniciamos a familiarización recorrendo ao calculo do segmento máis curto entre varios aleatorios, <http://dev.geogebra.org/trac/wiki/Jython>, no que aprendemos a conectar e crear puntos de Geogebra en Python, $geo.A=(1,2)$ ou $\$A=(1,2)$, despois crea 8 puntos de posición aleatoria e calcula o segmento máis pequeno dos que unen A cos outros puntos. Ao mover o punto A só queda visible o segmento máis curto. Como cambiaríamos máis curto por máis longo?

Cambiando *min* por *max* na función que se define.

Advertir que, como Geogebra 5 está en versión beta, é preciso facer cambios respecto ao texto na web. Se non poñemos *onUpdate*, exactamente así, con esta sintaxe, non funcionará.

Outras posibilidades:

Simetría. Ao mover o punto A creamos $\$B=Point(\$A.y,\$A.x)$ no evento onUpdate e nas propiedades dámoslle rastro a A e a B. Simetría respecto ao eixe y? Faríamos $\$B=Point(-\$A.x,\$A.y)$

Folla de cálculo. Este programa pon cores distintas segundo o número sexa o positivo ou negativo

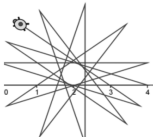
```
for r in range(1,6):
if  $\$[1,r]<0$ :
 $\$[1, r].color = Color.RED$ 
else:
 $\$[1, r].color = Color.BLACK$ 
```

Ou Lissajous

```
a = 2
b = 3
for s in range(100):
t=s*6/100
x = a * sin(2*t)
y = b * cos(3*t)
 $\$A=Point(x,y)$ 
```

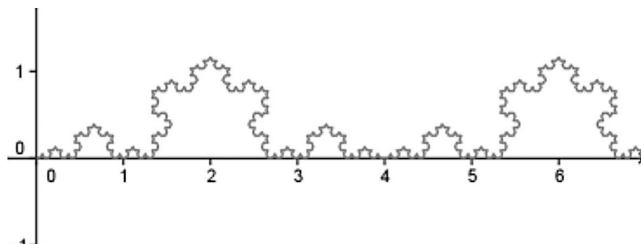
Programas en linguaxe Logo.

Na versión actual de Geogebra 5 os programas Logo deben acabar en `ggbApplet.startAnimation()`

<pre>t = Turtle() t.pen_color = Color.BLUE for _ in range(20): t.forward(4) t.turn_left(162)</pre>	
<pre>t=Turtle() t.pen_color=Color.RED for i in range(10): for j in range(5): t.forward(5) t.turn_left(72) t.forward(10) t.turn_left(108)</pre>	<pre>t=Turtle() t.pen_color=Color.RED for i in range(10): for j in range(5): t.forward(5) t.turn_left(108) t.forward(10) t.turn_left(72)</pre>

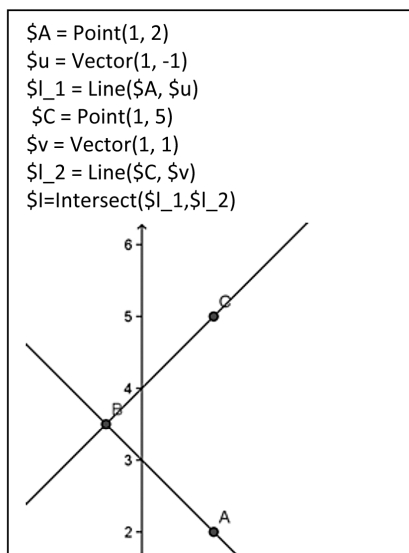
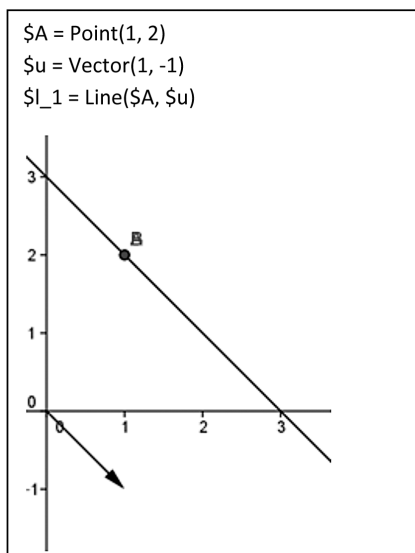
Pequenos cambios nos programas dan resultados gráficos notablemente distintos, como ao cambiar `t.forward(10)` por `t.forward(5)`

Fractais. Como temos a posibilidade de facer funcións recursivas, podemos representar fractais, como a curva de von Koch



Ao igual que esta curva, podemos reutilizar numerosos exemplos de gráficas tanto de programas Logo tradicionais como de adaptación de coleccións de exercicios de Python Logo.

Xeometría analítica do plano na ventá Python. Como vemos a continuación, en Python definimos unha recta con `Line` usando un punto e un vector. Geogebra dános o resultado en ecuación implícita e presenta a imaxe, permitindo pasar directamente dos enunciados dalgún tipo de problemas ás solucións.

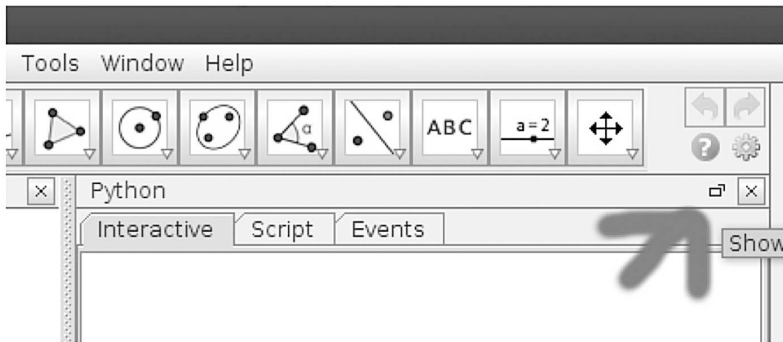


Na ventá Python de Geogebra podemos referirnos a valores numéricos dados por un esvarador ou deslizador (no exemplo que segue, a e b) e que polo tanto podemos cambiar, vendo como evolucionan os problemas en función dos parámetros, como neste exemplo, `c=Circle(($a,$a),5)`

```
l=Line((0,0),(2,2*$b))
p,q=Intersect(c,l)
```

Na versión Geogebra 5 podemos tratar figuras en 3D, por exemplo os poliedros platónicos. Tratar este punto precisaría máis tempo.

Unha apreciación necesaria é saber como se executa un script Python en Geogebra 5. É algo que tamén cambiou desde xullo a outubro 2012. Débese pinchar no símbolo da esquina superior dereita da ventá Python, tal como se sinala na imaxe, para que apareza a opción Run Script no menu que se desprega.



As múltiples vantaxes que xa coñecemos de Geogebra, -permite visualizar conceptos abstractos; facilita a súa representación, os estudantes poden facer conexións e experimentacións...- engádense agora outras moitas ao integrar Python con esta versión 5, xa que podemos ademais:

Iniciar a programación en Secundaria. Pódese aprender a facer pequenos programas con Python, iniciarse na programación e na sintaxe de “indentación” de Python que a fai máis lexible, e na utilización dos miles de módulos de Python existentes que resollen milleiros de tipos de problemas distintos.

Conectar datos externos con Geogebra. Con Python e ExpEyes (<http://expeyes.in>) podemos realizar medidas de experimentos en electricidade, magnetismo, son, electrónica, mecánica, óptica e calor.

Traballar na liña de GeogebraSTEM

http://orbit.educ.cam.ac.uk/wiki/GeoGebraSTEM_exploration_day atópanse varios exemplos de proxectos para modelizar situacións físicas.

Noutro sentido, **Raspberry Pi**, (a Pi é de Python), un ordenador minúsculo e moi barato, pódese usar para aprender a programar, como o propón a U. de Oxford, ver <http://www.raspberrypi.org/> e, tamén, usalo con Geogebra.