

# Programando con Robots y Python

Grupo de Desarrollo Lihuen  
soportelihuen@linti.unlp.edu.ar  
LINTI - Facultad de Informática - Universidad Nacional de La Plata

10 de septiembre de 2012

## PRACTICA 1

*El objetivo de esta actividad es instalar y utilizar las herramientas necesarias para trabajar con los robots y Python y conocer las funciones básicas de los robots. Como la plataforma de trabajo será Linux, también se trabajará sobre distintos aspectos de su uso, a medida sea necesario.*

### Algunos datos útiles

- Hay que actualizar los repositorios para agregar el paquete que maneja el robot.
- La especificación del hardware está en: <http://www.robotgroup.com.ar/web/>
- La documentación del curso se encuentra en <http://robots.linti.unlp.edu.ar>
- Dirección de contacto: [soportelihuen@info.unlp.edu.ar](mailto:soportelihuen@info.unlp.edu.ar)
- Responsable del Proyecto: Claudia Banchoff - [cbanchoff@info.unlp.edu.ar](mailto:cbanchoff@info.unlp.edu.ar) - @clauBanchoff

### Sobre Python

**Ejercicio 1.** Accediendo desde Lihuen, acceda al intérprete de Python y verifique qué versión tiene instalada

**Ejercicio 2.** Accediendo desde Lihuen, utilice el IDE Geany y testee la instalación. Genere un programa que muestre algún mensaje en la pantalla (por ejemplo: `print "hola mundo"`) y ejecútenlo desde el IDE y desde la consola.

### Primeros Pasos con el Robot

**Ejercicio 3.** Enumere las aplicaciones y librerías que debe tener instaladas en su máquina para comenzar a trabajar. Verifiquen cuáles están instaladas y cuáles no. Si realiza el taller con su propio equipo, instale las que hagan falta para comenzar a trabajar.

🔗 Repositorio oficial:

```
deb http://repo.lihuen.linti.unlp.edu.ar/lihuen squeeze/lihuen4 main contrib non-free
```

🔗 El paquete se llama: **robot**.

🔗 Para instalar, agregar el repositorio al archivo `/etc/apt/source.list`

🔗 Ejecutar con privilegios de administrador:

```
apt-get update
apt-get install robot
```

**Ejercicio 4.** Abra el Gestor de paquetes Synaptic y verifique las actualizaciones anteriores.

**Ejercicio 5.** Conecten el robot a la computadora. ¿Cómo nos damos cuenta que el robot está conectado? ¿Le ponemos un nombre?.

**Ejercicio 6.** Pruebe los comando para que el robot avance durante 3 segundos a velocidad máxima y luego retroceda a la misma velocidad.

**Ejercicio 7.** Pruebe con distintos argumentos las funciones de movimiento de manera tal que el robot pueda girar  $90^\circ$  y  $45^\circ$ .

**Ejercicio 8.** Genere un archivo denominado "giro45.py" y "giro90.py" donde queden guardadas las invocaciones donde el robot consigue girar el ángulo buscado. ¿Qué debe tener en cuenta para que esto funcione siempre así?

**Ejercicio 9.** Ejecute desde la consola los programas anteriores.

**Ejercicio 10.** Defina dos funciones Python denominadas giro45() y giro90() y guárdelas en un archivo denominado "giros.py"

**Ejercicio 11.** Escriba un programa que importe el módulo generado anteriormente y realice 5 giros a  $45^\circ$  y otros 6 a  $90^\circ$ .

**Ejercicio 12.** Intente que el robot emita una melodía. Prueben distintas frecuencias de sonidos con distintas duraciones. Guarde en un archivo denominado "melodia.py" la secuencia generada.