El debugger

El depurador (o debugger) es una herramienta que nos permite correr el programa paso a paso para encontrar el error (o bug) que tiene nuestro programa.

Cuando identificamos que hay un error en nuestro programa, tenemos una idea de los lugares donde podría estar el error. Entonces lo que hacemos es definir un punto de interrupción (breakpoint) para poder analizar las variables y seguir paso a paso cómo se va ejecutando el programa, si está siguiendo el flujo de ejecución que debería seguir o si las variables están teniendo los valores correspondientes.

PDB

La biblioteca estándar de Python provee un módulo para depurar el código llamado pdb. Para definir un *breakpoint* con pdb primero tenemos que importar el módulo y luego llamar a la función *set_trace*:

```
import pdb
pdb.set_trace()
```

En la línea que hayamos llamado a la función *set_trace*, se detendrá el programa y podremos tomar control de la ejecución del programa.

A continuación se presenta una breve referencia de los comandos más utilizados de pdb, estos comandos se utilizan cuando el programa frena en el punto que pusimos el *breakpoint* y nos da el control del mismo.

Comando	Descripción
h(elp)	Muestra la ayuda de todos los comandos que provee pdb.
s(tep)	Ejecuta la línea actual. Si en la línea actual hay una llamada de función se mete dentro del código de la función para poder seguir la ejecución de la misma.
n(ext)	Continúa la ejecución hasta la próxima línea.
r(eturn)	Continúa la ejecución hasta el return de la función actual.
c(ont(inue))	Continúa la ejecución hasta le próximo breakpoint. Si no hay otro breakpoint. Ejecutará hasta el fin del programa.

l(ist)	Muestra el código del archivo actual.
p	Evalúa la expresión pasada como parámetro, la evalua en el contexto actual e imprime su valor.
q(uit)	Sale del depurador. El programa es abortado.

Cabe destacar que se puede instalar la librería ipdb, que es una versión mejorada del módulo pdb. Por ejemplo, tiene autocompletado y se pueden recorrer los comando ejecutados anteriormente con la flecha hacia arriba.

Configuración del depurador en Visual Studio Code

En esta sección se explicará cómo correr y configurar el depurador de Visual Studio Code con Python.

Para definir un breakpoint nos posicionamos en la línea donde queremos definir el mismo y presionamos F9 o hacemos click a la izquierda donde esta el circulo rojo.

Para configurar el debugger seleccionar la configuración dentro de la caja de herramientas “debugger” o eligiendo la opción del menú Debug -> Open configurations.

Esto creará el archivo launch.json conteniendo varias configuraciones posibles.
Seleccionar la opción “Python: Current File (Integrated Terminal)”.

Luego presionar F5 o ir elegir del menú Debug -> Start Debugging para ejecutar el programa hasta el breakpoint definido.

Para más detalle del uso y configuración del depurador en Visual Studio Code ver la documentación[1].

[1] https://code.visualstudio.com/docs/python/python-tutorial#_configure-and-run-the-debugger.