

Trabajo Práctico #2: EvoPy

Entrega: Miércoles 9 de Noviembre

Se desea construir un generador de casos de tests para el lenguaje Python usando algoritmos genéticos. Su implementación debe ser capaz de generar invocaciones a las funciones definidas en un módulo python. Este módulo es indicado usando una ruta a un archivo con extensión .py.

Para simplificar vamos a suponer que los programas bajo análisis no contendrán declaraciones de clases. También vamos a ignorar todas las funciones cuyo nombre comience con un "_". Para implementar EvoPy puede seguir el pseudo-código del algoritmo genético implementado en EvoSuite:

INPUT: filePy, maxGenerations, maxTime, outputDir, random_seed

Algorithm 1 The genetic algorithm applied in EVOSUITE

```
1 current_population  $\leftarrow$  generate random population
2 repeat
3   Z  $\leftarrow$  elite of current_population
4   while  $|Z| \neq |current\_population|$  do
5     P1, P2  $\leftarrow$  select two parents with rank selection
6     if crossover probability then
7       O1, O2  $\leftarrow$  crossover P1, P2
8     else
9       O1, O2  $\leftarrow$  P1, P2
10    mutate O1 and O2
11    fP = min(fitness(P1), fitness(P2))
12    fO = min(fitness(O1), fitness(O2))
13    lP = length(P1) + length(P2)
14    lO = length(O1) + length(O2)
15    TB = best individual of current_population
16    if fO < fP  $\vee$  (fO = fP  $\wedge$  lO  $\leq$  lP) then
17      for O in {O1, O2} do
18        if length(O)  $\leq$  2  $\times$  length(TB) then
19          Z  $\leftarrow$  Z  $\cup$  {O}
20        else
21          Z  $\leftarrow$  Z  $\cup$  {P1 or P2}
22      else
23        Z  $\leftarrow$  Z  $\cup$  {P1, P2}
24    current_population  $\leftarrow$  Z
25 until solution found or maximum resources spent
```

- filePy es la ruta al archivo python con el módulo bajo test. Es una ruta absoluta (ejemplo: /home/johndoe/autotest/tp1/examples/arrays.py)

Generación Automática de Casos de Test - 2016

Ejercicios

- `maxGenerations`: es la cantidad máxima de generaciones permitidas para el algoritmo genético
- `maxTime`: es el tiempo máximo (en segundos) permitidos para ejecutar EvoPy
- `outputDir`: es la ruta (absoluta) a la carpeta donde se almacenarán los tests generados
- `random_seed`: es la semilla para configurar el generador aleatorio de Python

Para facilitar su trabajo, se cuenta con clases desarrolladas para el TP1 que permiten las siguientes funcionalidades:

- **TestCall**: representa un Test. Ya que sólo trabajaremos con módulos con funciones python, los test call son únicamente una invocación a una función python con sus argumentos. Un TestCall está compuesto por el nombre de la función a invocar y la lista de argumentos (TestArgument) a usar.
- **TestArgument**: son los argumentos (ie valores de parámetros) con que se necesitan para definir un TestCall.
- **TestExecutor**: ejecuta un TestCall.
- **TestWriter**: permite volcar el contenido de un TestCall en un archivo .py para su ejecución posterior como un unit test.

Además puede utilizar el Generador Aleatorio RandPy desarrollado en el TP1.

Parámetros

El algoritmo genético debe respetar los siguientes parámetros para su configuración

- **populationSize**: cantidad de individuos en la población (default 50)
- **eliteSize**: tamaño de la elite (default 1)
- **testSuiteLength**: tamaño máximo de un test suite (default 40)
- **crossoverProbability**: [0,1] probabilidad de aplicar crossover al seleccionar individuos (default 0.75)
- **mutationProbability**: [0,1] probabilidad de aplicar mutation en el nuevo individuo (default 0.75)
- **addNewTestProbability**: [0,1] probabilidad de insertar un nuevo test generado aleatoriamente (default 0.33)
- **modifyExistingTestProbability**: [0,1] probabilidad de modificar un test existente aleatoriamente (default 0.33)
- **removeExistingTestProbability**: [0,1] probabilidad de eliminar un test existente (default 0.33)

Evaluación

Una vez terminado el desarrollo de EvoPy debe evaluar su desempeño completando la siguiente tabla:

Example	Avg. Line Cov.	Avg. Branch Cov.
arrays.py		
cgi_decode.py		
convexhull.py		
coord.py		
encryption.py		
levenshtein.py		
persons.py		
sets.py		
sorting.py		
triangle.py		
tuples.py		
years.py		

Para ello, deberá ejecutar cada uno de los módulos de la Tabla usando EvoPy con la siguiente configuración:

- maxTime=120 (2 minutos)
- maxGenerations=-1(sin cota de generaciones)

Dado que la efectividad de EvoPy puede variar aleatoriamente, debe repetir la ejecución 10 veces usando distintas semillas de aleatoriedad (random_seed)