



Práctica 1.

El objetivo de esta práctica es implementar un algoritmo genético clásico para hallar el máximo o mínimo de diferentes funciones.

a) Diseño del algoritmo.

- **Representación de los individuos:** se representan mediante cadenas binarias que se corresponden con los puntos del espacio de búsqueda.
- **Función de evaluación:** es el resultado de evaluar la función considerada en el punto que resulta de la decodificación del individuo.
- **Selección:** por ruleta, torneo y otros métodos de forma opcional.
- **Operador de cruce:** Se selecciona un punto de cruce aleatoriamente y se intercambian los segmentos resultantes.
- **Operador de mutación:** Se cambia el valor del punto considerado para mutación.

Consideramos la optimización de las siguientes funciones:

▪ Función 1:

$$f(x) = x + \left| \sin(32\pi x) \right| : x \in [0,1]$$

que presenta un máximo de 1.98442447 en 0.98447395

▪ Función 2:

$$f(x,y) = 21.5 + x.\sin(4\pi x) + y.\sin(20\pi y) :$$
$$x \in [-3.0,12.1]$$
$$y \in [4.1,5.8]$$

que presenta un máximo de 38.809 en 11.625 y 5.726

▪ Función 3:

$$f = \frac{\sin x}{1 + \sqrt{x} + \frac{\cos x}{1+x}} : x \in [0, 25]$$

que presenta un mínimo de -0.318071 en 4.5798

▪ Función 4:

$$f(x_i, i=1..2) = \left(\sum_{i=1}^5 i \cdot \cos((i+1)x_1 + i) \right) \left(\sum_{i=1}^5 i \cdot \cos((i+1)x_2 + i) \right)$$
$$x_i \in [-10,10]$$

que presenta 18 mínimos de -186.7309

▪ **Función 5:**

$$f(x_i | i = 1..n) = - \sum_{i=1}^n \sin(x_i) \sin^{20} \left(\frac{(i+1)x_i^2}{\pi} \right) : x_i \in [0, \pi]$$

que presenta los siguientes mínimos en función de n :

n	1	2	3	4	5	6	7	8
mínimo	-1	-1.959091	-2.897553	-3.886358	-4.886358	-5.879585	-6.862457	-7.858851

b) Interfaz gráfica: Ajuste interactivo de los parámetros del algoritmo.

- ❑ La aplicación debe permitir variar los parámetros interactivamente. Los parámetros que pueden especificarse son: tamaño de la población, número límite de iteraciones del algoritmo, porcentaje de cruces, porcentaje de mutaciones y precisión o valor de error para la discretización del intervalo.

c) Representación gráfica de evolución

- ❑ Representación gráfica en la que se señale el máximo o mínimo encontrado por el algoritmo y los valores medio y máximo de la aptitud a lo largo de las generaciones para estudiar su evolución. (herramientas *jmathtools*, *jfreechart*...)



d) Parte opcional

- ❑ Elitismo: permite introducir elitismo en el algoritmo, garantizando la supervivencia de un pequeño porcentaje (especificado como un parámetro) de la población, formado por los mejores individuos.
- ❑ Análisis y estudio de los resultados obtenidos al ir variando ciertos parámetros dentro de un rango de valores. Para ello fijamos todos los parámetros menos uno que vamos variando para realizar el estudio. Se incluirán en la memoria.

Documentación a entregar

- ❑ Hay que enviar al campus virtual antes del **26 de Marzo a las 12:00** un archivo comprimido con el código java de la aplicación (**proyecto en Eclipse** cuyo nombre se corresponde con el nombre del grupo y las siglas P1, por ejemplo **G03P1**). Es importante seguir esta notación. En el campus virtual el ejercicio está identificado como **Practica 1**. También se debe incluir un archivo con la memoria que debe incluir un estudio resumido de las gráficas y del resultado obtenido con cada función.
- ❑ El **día de corrección en Laboratorio (27 de Marzo)** deberán estar presentes los dos miembros del grupo para defender el trabajo realizado.