



Centro Universitario de Ciencias Exactas e Ingenierías

Asignatura: **Seminario Inteligencia Artificial**

Sección: D04

Actividad 8

Alumno: Luis Jaime Portillo Correa

Código: 217546155

Profesor: **Javier Enrique Gómez Ávila**

Fecha: **26/10/2023**

Resultados Función McCormick

Para el desarrollo de la práctica con la función McCormick haremos uso del algoritmo de funciones de penalización, esto porque anteriormente pudimos notar que la mayoría de los algoritmos daban problemas con esta función específicamente, para ser más claros podemos decir que al momento de realizar las iteraciones, el rango de búsqueda se perdía y terminaba sin darnos ninguna señal de algún mínimo global o local.

Prácticamente se hace uso del algoritmo de evolución diferencial, pero en esta ocasión se utilizan funciones de penalización que regularán el rango de búsqueda de manera que el algoritmo no se pierda a lo largo de las iteraciones, para comenzar utilicé estos valores iniciales como parámetros:

```
Parámetros

✓ 0s ▶ x1 = np.array([-1.5, -3])
      xu = np.array([4, 4])

      G = 30
      N = 50
      D = 2

      F = 0.6
      CR = 0.9

      x = np.zeros((D, N))
      fitness = np.zeros(N)
```

Una vez definida la función base y la función con penalización, podremos comenzar a calcular los resultados deseados.

```
Función objetivo

✓ 0s [4] f = lambda x, y: np.sin(x+y) + (x-y) ** 2 - 1.5 * x + 2.5 * y+1 # McCormick Function
      fp = lambda x, x1, xu: f(x[0], x[1]) + 1000 * Penalty(x, x1, xu);
```

En esta captura podemos observar que los resultados son bastante buenos, prácticamente coinciden con lo que esperábamos como valor de la función y como mínimo global.

Finalmente, el mínimo global se encuentra en:

$$f(x^*, y^*) = -1.9133, x = -0.54719, y = -1.54719$$

