Missouri State University
Department of Computer Science

Assignment 2: Evolutionary Computing
Summer 2025

Submission Instructions

Students must follow these submission guidelines to receive full credit:

1. **Submission:**
   - Submit a single compressed '.zip' file containing all source files (Python scripts) and PDF report for conceptual questions.
2. **File Naming Convention:** Name your '.zip' file as: `FirstName_LastName_EC_AS2.zip`
3. **Deadline:** Submit your assignment by June 24^{th} 2025.
4. **Academic Integrity:**
   - Your submission must be your own work.
   - Plagiarism will result in a zero grade and possible disciplinary action.

Objective

The purpose of this assignment is to implement and analyze Genetic Algorithms (GA) using **binary encoding** to optimize the well-known **De Jong test functions**. You will write code to:

- Initialize binary chromosomes
- Decode them to real values
- Evaluate De Jong functions
- Apply GA operators (selection, crossover, mutation)

De Jong's Test Functions

De Jong proposed five functions as standard benchmarks for continuous function optimization. You are required to work with the following:

1. **Sphere Model:**

$$f_1(\mathbf{x}) = \sum_{i=1}^{n} x_i^2 \quad \text{with } x_i \in [-5.12, 5.12]$$

2. **Weighted Sphere Model:**

$$f_2(x) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2 \quad \text{with } x_i \in [-2.048, 2.048]$$

3. **Step Function:**

$$f_3(\mathbf{x}) = \sum_{i=1}^{n} [x_i] \in [-5.12, 5.12]$$

   choose n=4 for this function.

4. **Noisy Quartic:**

$$f_4(\mathbf{x}) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0, 1) \quad \text{with } x_i \in [-1.28, 1.28]$$

   choose n=4 for this function.

Assignment Tasks

Q1. Binary Encoding and Initialization

- Encode each variable using binary strings (suggested: 8-bit and 16-bit).
- Implement a function to generate a random initial population of binary chromosomes.

Q2. Chromosome Decoding and Function Evaluation

1. Write a function to decode a binary chromosome into a real number using linear mapping to the domain of each De Jong function.
2. Implement evaluation functions for **all** of the De Jong functions listed above.

Q3. GA Operations

Implement the following Genetic Algorithm operations:

1. Fitness Proportionate Selection (Roulette Wheel)
2. One-point or Two-point Crossover (probability = 0.90)
3. Bitwise Mutation (probability = 1/Length)

Q4. GA Execution

- Run your Genetic Algorithm for 50 generations with a population size of 20. (Change population size and number of generation if not able to find good solutions, but don't go beyond 50 for both.)
- Plot the best fitness and average fitness per generation for each function.
- Report the best solution found and its decoded real values.

Q5. Analysis and Comparison

1. Compare the convergence behavior of all functions.
2. Which function was easiest to optimize? Which was hardest? Explain why.