AI—PROJECT II
Genetic Algorithm

1. Following the example in the Lecture Notes, define your own function $f(x)$ whose maximum you want to find. Make sure the function has more than two peaks so that, apart from the global maximum, it has also local maxima. Consider also using irrelevant bits, in the chromosome.

2. Create a population where each specimen, x, is expressed as a binary string of sufficient length (say, 20–30 bits). On this initial population, run the genetic algorithm with the usual operators: survival, recombination (2-point crossover), and mutation. Give some thought to how to handle the termination criterion.

3. Run systematic experiments to learn more about the genetic algorithm's behavior under diverse conditions. The following performance characteristics are of special interest.

   - How does the fitness of the population's best specimen (and the average fitness of the population) evolve in time?
   - How does this fitness improvement depend on mutation rate?
   - How does the number of generations needed to find the function's maximum depend on the size of the population?
   - What is the total number of individuals seen before the solution has been found (i.e., how many times has $f(x)$ been calculated)? How does this number depend on the size of the population?
   - Did you observe "premature degeneration"? How did you handle it?

4. Write a report of about 5 pages, specifying the function $f(x)$ and the research objectives, describing the experiments, and showing graphs that illustrate the program's behavior. Discuss the results and the lessons learned from these results. Make sure the report is appropriately structured into sections—the more it resembles a scientific publication, the better.