Generational Genetic Algorithm

Tyler Wilson 100773241

This application demonstrates genetic algorithms by generating 6 random “people” with 8 bit and creating new generations of their “offspring”.

1) We initialize a 1st generation by randomly generating 6 numbers between 0 and 254.
2) We converted those numbers to 8 bit decimals to represent chromosomes (person)
3) For each “person”, we calculated the fitness of their chromosomes to weight against the other people
4) A “roulette” function is called to calculate the individuals chances of being chosen for mating.
5) 6 people with repetition are selected to be paired with each other, to make 3 couples.
6) Each couple will create two new chromosomes based on cross over odds and mutation odds.
    a. If the odds are in favour of cross over, the new chromosome takes the father’s 4 bits on the left and the mother’s 4 bits on the right.
    b. The children are assigned genders at random and given a traditional first name for that gender.
    c. The childrens’ last name becomes same last name as the father position
    d. If the odds are for mutation, one randon bit in the child’s chromosome is switched to it’s opposite value.
7) The next generation is filled with the children of the previous generation.
8) You can adjust how many generations to make by adjusting the GENERATIONS constant in the app.py file.
9) The app will create a new output to a csv file everytime you run it. The filenames are unique and based on datetime.now() and my credentials.