Homework 1 – Computational Intelligence

Frederik Darwin

M10601836

- Assigned Optimization Problem:

Problem 4

Maximize $f(x,y) = 8 - \frac{\sin^2(\sqrt{(x^2+y^2)})}{(1+0.001*(x^2-y^2))^8}$ $-1 \leqq x \leqq 2$; $-1 \leqq y \leqq 1$; $x+y \geqq -1$

Maximum=8 at (x,y)=(0,0)

- With the precision of 10^{-3}, x and y have a range of $3x10^3$ and $2x10^3$ respectively. Looking at this number, we can conclude that:

$$2^{10} = 1024 < 2000 < 2^{11} = 2048 < 3000 < 2^{12} = 4096$$

- X needs 12 and Y needs 11 Chromosome, totaling 23bit of genes per individual. Generation that I use is only 100, with 20 population per generation. But the number can be easily changed as I use it as a global variable.
- Selection mechanisms used is roulette-wheel, where more fit the individual, more chance the individual will selected by the roulette to be the parent. There are 3 parents that I choose, where the 1st individual chosen by the selection system will mate with 2nd and 3rd individual chosen by selection system too. Two of the least fit individuals in the population already discarded from the selection system, preventing them to mate and creating bad genes.
- Crossover that I use is a simple one, a one-point cross-over. The probability of the crossover and where it's point will be is already decided by the selection system. From four children created by 1st, 2nd, 3rd individual mating, only two will be selected by which one is the most fit. Children from 1st and 2nd will fight to the dead to get its 49th position in the next gen population, and children from 1st and 3rd for 50th position. (2 least fit individual from last generation already discarded).
- Tuning parameter that I use are: crossover rate = 0.9, and mutation rate = 0.1. It can also be easily changed as I use it as global variable.
- The fittest gen that I get from the GA code is 7.999, it is pretty much the same as answer that the problem seeks.

```
In [16]:  1  fittestnextgen[-1:]

Out[16]: [7.999]
```

Out[13]: <matplotlib.legend.Legend at 0x130011b0>