Part 1
Download go_ga.m, blackbg.m, evalpopu.m, evaleach.m, bit2num.m, peaksfcn.m, nextpopu.m from the
books web site. Experiment with different parameter settings; population sizes (10, 100), crossover rates
(10%, 80%), mutation rates (0.01%, 5% per bit), and generations (10, 500).
To make the code run under matlab version 6 (you can check which version you have by typing “version”
at the matlab prompt), change the following lines in the peaksfcn.m by adding the red statement
“matlabv==6”:
```matlab
if matlabv==4,
    property='linestyle';
elseif matlabv==5 | matlabv==6,
    property='marker';
else
    error('Unknown MATLAB version!');
end
```
Part 2
Modify go_ga.m to create a Genetic Algorithm to solve the minimum
of the banana function
\[ f(x) = 100(y-x)^2 + (1-x)^2 \]
in the range -2<x<2 and -1<y<3.
Note: you do not need to plot the contours as is done in the go_ga for the peaks function.
Create a binary chromosome that represents all possible x and y values to 4 digits after the decimal point.

What to hand in:

Part 1
Describe what happened when you changed the parameter settings. Did it always act the same with the
same settings? (4 points)
Part 2
Describe the chromosome you used to represent the parameters to the banana function (how many bits does
it have and what is their meaning). (3 points)
What did you use for the population size and number of generations, and why? (3 points)
What did you use for crossover and mutation rates, and why? (3 points)
What is the minimum point? (2 points)
For one run of your GA print out the following
1) the x, y, and fitness of the best member of each generation (2 points)
2) the entire population after generations 5 and 10. (2 points)
3) a plot with the fitness of the best, average, and worst item for each generation. (2 points)
Your code for the banana GA (emailed to us). (4 points)