Homework 3 - Local Search
100pts + 10pts bonus

Goal: To learn more about local search which has very low memory usage and can be quite successful for many problems; and to gain further experience with programming and reporting research results.

Task Overview: The task is to solve the N-queen problem with different local search algorithms. N-queen problem is finding the placement of N queens on an $N \times N$ board such that none of the queens one attacks another. The given Python code implements basic hill climbing and random restarts. You will be asked to run simulation experiments with the given code and expand the code with stochastic hill climbing and simulated annealing.

In more detail, answer/complete the following and fill the appropriate rows in Table 1.

a) (40 points) Given the code in link[1], run 100 simulations/experiments with different initial solutions and fill in Table 1 with the number of successes, number of iterations, and the time it takes to find the solution on average for N=20, for basic **Hill Climbing** and **Hill Climbing with Random Restart**s with two values for the number of restarts (k=10, 100).

b) (40 points) Add a new function $randomNeighbor(\ldots)$ and necessary code to implement **Stochastic Hill Climbing** and fill the results of 100 experiments to the corresponding row of Table 1. If there is no better neighbor than the current state, the function should return the current state. Leave the rest of the code unchanged, but make sure to comment any new code you add.

c) (10 points) Enter your Colab link of your code or leave blank if you did not add any new code! Important: Make a copy of the given Colab file first, complete (add your name in the beginning of the file), test and then share your link in **editable** format. If you do not include a proper and editable link, you will get zero; so, check that someone else can access your link (you can also try in anonymous mode).

d) (10 points) Based on your observations, which of the three hill climbing variations (basic, stochastic or random restart) performed well - just summarize in 1-2 sentences.

e) (Bonus - 10 points) Implement **simulated annealing** and fill the results of 100 experiments to the corresponding row of the Table 1. You should start with a high temperature, such as $T = 10000$ and implement a cooling period with cooling rate such as $\alpha = 0.95$, where you update the temperature by a factor of α at each iteration. Play around to see if you can find better parameters. *You can use* $random.uniform(0, 1)$ *in deciding if a worse neighbor should be accepted.*

What to Submit: A pdf document with your filled table. You can use Overleaf for editing a given LaTeX template[2]. Alternatively, you can prepare your table in MS Word etc. and generate a pdf document. Handwritten, hand-filled tables will lose 10 pts.

[1] https://colab.research.google.com/drive/1XmWQ-Pq2n5wvhe7WkVR4ejgRW124IPYJ?usp=sharing
[2] https://www.overleaf.com/read/zjnmgscvchcpa3d3c2

Table 1: Simulation results. You will submit a Pdf of this page.

	N=20		
	Percentage of success in 100 runs	Elapsed time to complete experiment (secs)	Solutions found in how many restarts on average
Basic Hill Climbing			-
Random Restart with k=10			
Random Restart with k=100			
Stochastic Hill climbing			-
Simulated Annealing if implemented ($T = \ldots$ and $\alpha = \ldots$)			-
c) Colab link for your solution			
d) Enter your short summary			