INF273 – Assignment #3

Implement a Random Search, a Local Search, and a Simulated Annealing algorithm to solve a pickup and delivery problem with time windows.

- Five test instances are given (you can find them at mitt.uib under the data files)
- You should implement the following algorithms, run them 10 times for each instance and report five tables (one for each problem instance) in the given format
- Start with an initial solution that every call is outsourced
- You should also report the best found solution for each instance (in the same format as it has been discussed in the class and is available on the lecture slides)
- You should use *2-exchange*, *3-exchange*, and *1-reinsert* heuristics as operators.
- Improvement (%) = 100 * (Objective of Initial solution - Best objective) / Objective of Initial solution
- Tune the parameters!
- You should read above bullet points one more time!

Instance name (e.g. Call_7_Vehicle_3)				
	Average objective	Best objective	Improvement (%)	Running time
Random Search				
Local Search				
Simulated Annealing				

Local search (modified for assignment #3)

Input: initial solution (s_0),
Input: neighborhood operators (*2-exchange*, *3-exchange*, and *1-reinsert*)
Parameters: P_1 (probability of using *2-exchange*), P_2, P_3 ($=1 - P_1 - P_2$)
Input: evaluation function f, $f(s) \longrightarrow$ the cost of s
$Current \Leftarrow s_0$, $BestSolution \Leftarrow s_0$
for iteration = 1 to 10000
    if $Rand < P_1$ then $Current \Leftarrow implement\ (2 - exchange)$on $BestSolution$
    elseif $Rand < P_1 + P_2$ then $Current \Leftarrow implement\ (3 - exchange)$on $BestSolution$
    else $Current \Leftarrow implement\ (1 - reinsert)$on $BestSolution$
    end if
    if $Current$ is feasible and $f(Current) < f(BestSolution)$ then
        $BestSolution \Leftarrow Current$
    end if
end for

Random search (modified for assignment #3)

Input: initial solution (s_0),
Input: evaluation function f, $f(s) \longrightarrow$ the cost of s
$BestSolution \Leftarrow s_0$
for iteration = 1 to 10000
    $Current \Leftarrow Generate\ a\ random\ solution$
    if $Current$ is feasible and $f(Current) < f(BestSolution)$ then
        $BestSolution \Leftarrow Current$
    end if
end for

Simulated Annealing (modified for assignment #3)

Input: initial solution (s_0),
Input: neighborhood operators (*2-exchange*, *3-exchange*, and *1-reinsert*)
Parameters: P_1 (probability of using *2-exchange*), P_2, P_3 ($=1 - P_1 - P_2$)
Parameters: T_0 (initial temperature), α (Cooling_factor)
Input: evaluation function f, $f(s) \longrightarrow$ the cost of s
$Incumbent \Leftarrow s_0$, $BestSolution \Leftarrow s_0$, $T \Leftarrow T_0$
for iteration = 1 to 10000
    if $Rand < P_1$ then $NewSolution \Leftarrow implement\ (2 - exchange)$on $Incumbent$
    elseif $Rand < P_1 + P_2$then $NewSolution \Leftarrow implement\ (3 - exchange)$on $Incumbent$
    else $NewSolution \Leftarrow implement\ (1 - reinsert)$on $Incumbent$
    end if
    $\Delta E \Leftarrow f(NewSolution) - f(Incumbent)$
    if $NewSolution$ is feasible and $\Delta E < 0$ then
        $Incumbent \Leftarrow NewSolution$
        if $f(Incumbent) < f(BestSolution)$ then
            $BestSolution \Leftarrow Incumbent$
        end if
    elseif $NewSolution$ is feasible and $RandII < p = e^{\frac{-\Delta E}{T}}$
        $Incumbent \Leftarrow NewSolution$
    end if
    $T = \alpha * T$
end for