Date __/__/__
Page ____

LAB - 5

Stimulated Annealing

Objective Function: $x^2 + 5\sin x$

```
function simulated Annealing (initial_state, cooling_state iteration,
initial temperature,
current state = initial_state
best_state = current_state
best_cost = objective function (current_state
temperature = initial_temperature
while temp>1:
        for i <- 1 to iteration
      new_state = Neighbour (current state)
    current_cost = objective function (curr_state)
  new_cost = objective function (new_state)

  if AP (curr_cost, new_cost, temp) >
          Random (0, 1)
      current_state = new_state
     if new_cost < best_cost
           best_state = new_state
          best_cost = new_cost
  temp* = cooling_rate
  return (best_state, best_cost)
  Function objective function (state):
        Cost = 0
     for ele in state
        Cost += $x^2$ + 5 sin x
  return cost

  function neighbour (state)
     new_state = state_copy()
    index = Random (0, length(state)-1)
  new_state [index] += Random (-1, 1)
_state,
teration)

urrent_state)

rrent state)
r_state)
tate)

}

e)-1)
1)
```

```
    return new state
  function A f Neighbours A P(curr_cost,
  new_cost, temp)
     if (new_cost < curr_cost);
         return 1
     else
         return e^(curr_cost - new_cost)/temp

  def main():
     initial_temp = 1000
     cooling_rate = 0.9
     iteration = 1000
     initial_state = [random.uniform(-10,
  10) for i in range (7)]
  best_state, best_cost = Simu(initial_state,
  initial_temp, cooling_rate, iteration)
  printf("Best state : {best state}")
  printf("Best cost : {best_cost}")
```

Output

```
Best state = [-0.2587, -0.13911, -0.1005,
     -0.0901, -0.2483, 0.0074, -0.1004)
Best cost = 0.17660
```