演算法 Homework 20191227
山下夏輝（Yamashita Natsuki）
R08922160

Mini HW #10

課題の提出

期限 金曜日 17.20 まで **点数** 1
提出しています テキスト入力ボックスまたはファイルのアップロード **ファイル タイプ** pdf
使用可能 2019年12月27日 日 17.20 ~ 1月3日 日 17.20 7 日

Consider another approximate algorithm for TSP with triangle inequality:

NEAREST-ADDITION (G)

1. Select the shortest edge e on the graph. let $S = \{u, \ v\}$, where u, v are the 2 end points of e. The initial tour T is uv.
2. While $| \ S \ | < \ N$:
   (a.) Find a node $v_j \in V \setminus S$, that is closest to S via v_i. let $S = S \cup \{v_j\}$
   (b.) update T: replace $v_i v_k$ with $v_i v_j v_k$
3. return T

Hear's an example: (https://imgur.com/yYFCUqD)

Please prove that ***NEAREST-ADDITION*** is a 2-approximation algorithm.

1.
Using MST

1. Select the starting point randomly
2. Get MST from the starting point
3. Add each costs of edges by the order of development of vertices
4. Return the cost

The cost is not more than double of MST. If and only if all vertices u, v, w in V, w have c(u, w) ≤ c(u, v) + c(v, w), the cost is not also more than double of TSP. The reason why the max cost is double of MST is that every edge in MST is passed through 2 times.

2.
As well as using MST algorithm, all vertices u, v, w in V, w have c(u, w) ≤ c(u, v) + c(v, w),
So c(u, w) - c(v, w) + c(u, v) ≤ 2c(u, v). Then total cost is not more than the sum of 2c(u, v) on every edges. Therefore, it is a 2- approximation algorithm.