

```

1. public class ClusteringVertex extends Vertex<TwoIntWritable, TwoIntWritable> {
2.
3.     @Override
4.     public void compute(Iterable<TwoIntWritable> messageValues, int superstepNo){
5.         if (superstepNo == 1) {
6.             Set<Integer> clusterCenters = getGlobalObjects("cluster-centers");
7.             setValue(clusterCenters.contains(getId()) ?
8.                 new TwoIntWritable(0, getId()) : new TwoIntWritable(Integer.MAX_VALUE, null));
9.             if (clusterCenters.contains(getId())) {
10.                 sendMessages(getNeighborIds(), getValue());
11.             }
12.         } else {
13.             int minDistance = getValue().value().fst;
14.             int minDistanceClusterId = getValue().value().snd;
15.             for (TwoIntWritable message : messageValues) {
16.                 if (message.value().fst < minDistance) {
17.                     minDistance = message.value().fst;
18.                     minDistanceClusterId = message.value().snd;
19.                 }
20.             }
21.             if (minDistance < getValue().value().fst) {
22.                 setValue(new TwoIntWritable(minDistance, minDistanceClusterId));
23.                 sendMessages(getNeighborIds(), getValue());
24.             } else {
25.                 voteToHalt();
26.             }
27.         }
28.     }
29. }

```