#include <iostream>

#include <vector>

#include <math.h>

#include <stdlib.h>

#include <time.h>

#include <algorithm>

using namespace std;

class Point

{

private:

int id\_point, id\_cluster;

vector<double> values;

int total\_values;

string name;

public:

Point(int id\_point, vector<double>& values, string name = "")

{

this->id\_point = id\_point;

total\_values = values.size();

for(int i = 0; i < total\_values; i++)

this->values.push\_back(values[i]);

this->name = name;

id\_cluster = -1;

}

int getID()

{

return id\_point;

}

void setCluster(int id\_cluster)

{

this->id\_cluster = id\_cluster;

}

int getCluster()

{

return id\_cluster;

}

double getValue(int index)

{

return values[index];

}

int getTotalValues()

{

return total\_values;

}

void addValue(double value)

{

values.push\_back(value);

}

string getName()

{

return name;

}

};

class Cluster

{

private:

int id\_cluster;

vector<double> central\_values;

vector<Point> points;

public:

Cluster(int id\_cluster, Point point)

{

this->id\_cluster = id\_cluster;

int total\_values = point.getTotalValues();

for(int i = 0; i < total\_values; i++)

central\_values.push\_back(point.getValue(i));

points.push\_back(point);

}

void addPoint(Point point)

{

points.push\_back(point);

}

bool removePoint(int id\_point)

{

int total\_points = points.size();

for(int i = 0; i < total\_points; i++)

{

if(points[i].getID() == id\_point)

{

points.erase(points.begin() + i);

return true;

}

}

return false;

}

double getCentralValue(int index)

{

return central\_values[index];

}

void setCentralValue(int index, double value)

{

central\_values[index] = value;

}

Point getPoint(int index)

{

return points[index];

}

int getTotalPoints()

{

return points.size();

}

int getID()

{

return id\_cluster;

}

};

class KMeans

{

private:

int K; // number of clusters

int total\_values, total\_points, max\_iterations;

vector<Cluster> clusters;

// return ID of nearest center (uses euclidean distance)

int getIDNearestCenter(Point point)

{

double sum = 0.0, min\_dist;

int id\_cluster\_center = 0;

for(int i = 0; i < total\_values; i++)

{

sum += pow(clusters[0].getCentralValue(i) -

point.getValue(i), 2.0);

}

min\_dist = sqrt(sum);

for(int i = 1; i < K; i++)

{

double dist;

sum = 0.0;

for(int j = 0; j < total\_values; j++)

{

sum += pow(clusters[i].getCentralValue(j) -

point.getValue(j), 2.0);

}

dist = sqrt(sum);

if(dist < min\_dist)

{

min\_dist = dist;

id\_cluster\_center = i;

}

}

return id\_cluster\_center;

}

public:

KMeans(int K, int total\_points, int total\_values, int max\_iterations)

{

this->K = K;

this->total\_points = total\_points;

this->total\_values = total\_values;

this->max\_iterations = max\_iterations;

}

void run(vector<Point> & points)

{

if(K > total\_points)

return;

vector<int> prohibited\_indexes;

// choose K distinct values for the centers of the clusters

for(int i = 0; i < K; i++)

{

while(true)

{

int index\_point = rand() % total\_points;

if(find(prohibited\_indexes.begin(), prohibited\_indexes.end(),

index\_point) == prohibited\_indexes.end())

{

prohibited\_indexes.push\_back(index\_point);

points[index\_point].setCluster(i);

Cluster cluster(i, points[index\_point]);

clusters.push\_back(cluster);

break;

}

}

}

int iter = 1;

while(true)

{

bool done = true;

// associates each point to the nearest center

for(int i = 0; i < total\_points; i++)

{

int id\_old\_cluster = points[i].getCluster();

int id\_nearest\_center = getIDNearestCenter(points[i]);

if(id\_old\_cluster != id\_nearest\_center)

{

if(id\_old\_cluster != -1)

clusters[id\_old\_cluster].removePoint(points[i].getID());

points[i].setCluster(id\_nearest\_center);

clusters[id\_nearest\_center].addPoint(points[i]);

done = false;

}

}

// recalculating the center of each cluster

for(int i = 0; i < K; i++)

{

for(int j = 0; j < total\_values; j++)

{

int total\_points\_cluster = clusters[i].getTotalPoints();

double sum = 0.0;

if(total\_points\_cluster > 0)

{

for(int p = 0; p < total\_points\_cluster; p++)

sum += clusters[i].getPoint(p).getValue(j);

clusters[i].setCentralValue(j, sum / total\_points\_cluster);

}

}

}

if(done == true || iter >= max\_iterations)

{

cout << "Break in iteration " << iter << "\n\n";

break;

}

iter++;

}

// shows elements of clusters

for(int i = 0; i < K; i++)

{

int total\_points\_cluster = clusters[i].getTotalPoints();

cout << "Cluster " << clusters[i].getID() + 1 << endl;

for(int j = 0; j < total\_points\_cluster; j++)

{

cout << "Point " << clusters[i].getPoint(j).getID() + 1 << ": ";

for(int p = 0; p < total\_values; p++)

cout << clusters[i].getPoint(j).getValue(p) << " ";

string point\_name = clusters[i].getPoint(j).getName();

if(point\_name != "")

cout << "- " << point\_name;

cout << endl;

}

cout << "Cluster values: ";

for(int j = 0; j < total\_values; j++)

cout << clusters[i].getCentralValue(j) << " ";

cout << "\n\n";

}

}

};

int main(int argc, char \*argv[])

{

srand (time(NULL));

int total\_points, total\_values, K, max\_iterations, has\_name;

cin >> total\_points >> total\_values >> K >> max\_iterations >> has\_name;

vector<Point> points;

string point\_name;

for(int i = 0; i < total\_points; i++)

{

vector<double> values;

for(int j = 0; j < total\_values; j++)

{

double value;

cin >> value;

values.push\_back(value);

}

if(has\_name)

{

cin >> point\_name;

Point p(i, values, point\_name);

points.push\_back(p);

}

else

{

Point p(i, values);

points.push\_back(p);

}

}

KMeans kmeans(K, total\_points, total\_values, max\_iterations);

kmeans.run(points);

return 0;

} /\*

[pict@localhost ~]$ g++ kmeans.cpp

[pict@localhost ~]$ ./a.out < dataset2.txt

Break in iteration 3

Cluster 1

Point 117: 6.5 3 5.5 1.8 - Iris-virginica

Point 53: 6.9 3.1 4.9 1.5 - Iris-versicolor

Point 78: 6.7 3 5 1.7 - Iris-versicolor

Point 101: 6.3 3.3 6 2.5 - Iris-virginica

Point 103: 7.1 3 5.9 2.1 - Iris-virginica

Point 104: 6.3 2.9 5.6 1.8 - Iris-virginica

Point 105: 6.5 3 5.8 2.2 - Iris-virginica

Point 106: 7.6 3 6.6 2.1 - Iris-virginica

Point 108: 7.3 2.9 6.3 1.8 - Iris-virginica

Point 109: 6.7 2.5 5.8 1.8 - Iris-virginica

Point 110: 7.2 3.6 6.1 2.5 - Iris-virginica

Point 111: 6.5 3.2 5.1 2 - Iris-virginica

Point 112: 6.4 2.7 5.3 1.9 - Iris-virginica

Point 113: 6.8 3 5.5 2.1 - Iris-virginica

Point 116: 6.4 3.2 5.3 2.3 - Iris-virginica

Point 118: 7.7 3.8 6.7 2.2 - Iris-virginica

Point 119: 7.7 2.6 6.9 2.3 - Iris-virginica

Point 121: 6.9 3.2 5.7 2.3 - Iris-virginica

Point 123: 7.7 2.8 6.7 2 - Iris-virginica

Point 125: 6.7 3.3 5.7 2.1 - Iris-virginica

Point 126: 7.2 3.2 6 1.8 - Iris-virginica

Point 129: 6.4 2.8 5.6 2.1 - Iris-virginica

Point 130: 7.2 3 5.8 1.6 - Iris-virginica

Point 131: 7.4 2.8 6.1 1.9 - Iris-virginica

Point 132: 7.9 3.8 6.4 2 - Iris-virginica

Point 133: 6.4 2.8 5.6 2.2 - Iris-virginica

Point 135: 6.1 2.6 5.6 1.4 - Iris-virginica

Point 136: 7.7 3 6.1 2.3 - Iris-virginica

Point 137: 6.3 3.4 5.6 2.4 - Iris-virginica

Point 138: 6.4 3.1 5.5 1.8 - Iris-virginica

Point 140: 6.9 3.1 5.4 2.1 - Iris-virginica

Point 141: 6.7 3.1 5.6 2.4 - Iris-virginica

Point 142: 6.9 3.1 5.1 2.3 - Iris-virginica

Point 144: 6.8 3.2 5.9 2.3 - Iris-virginica

Point 145: 6.7 3.3 5.7 2.5 - Iris-virginica

Point 146: 6.7 3 5.2 2.3 - Iris-virginica

Point 148: 6.5 3 5.2 2 - Iris-virginica

Point 149: 6.2 3.4 5.4 2.3 - Iris-virginica

Cluster values: 6.85 3.07368 5.74211 2.07105

Cluster 2

Point 45: 5.1 3.8 1.9 0.4 - Iris-setosa

Point 1: 5.1 3.5 1.4 0.2 - Iris-setosa

Point 2: 4.9 3 1.4 0.2 - Iris-setosa

Point 3: 4.7 3.2 1.3 0.2 - Iris-setosa

Point 4: 4.6 3.1 1.5 0.2 - Iris-setosa

Point 5: 5 3.6 1.4 0.2 - Iris-setosa

Point 6: 5.4 3.9 1.7 0.4 - Iris-setosa

Point 7: 4.6 3.4 1.4 0.3 - Iris-setosa

Point 8: 5 3.4 1.5 0.2 - Iris-setosa

Point 9: 4.4 2.9 1.4 0.2 - Iris-setosa

Point 10: 4.9 3.1 1.5 0.1 - Iris-setosa

Point 11: 5.4 3.7 1.5 0.2 - Iris-setosa

Point 12: 4.8 3.4 1.6 0.2 - Iris-setosa

Point 13: 4.8 3 1.4 0.1 - Iris-setosa

Point 14: 4.3 3 1.1 0.1 - Iris-setosa

Point 15: 5.8 4 1.2 0.2 - Iris-setosa

Point 16: 5.7 4.4 1.5 0.4 - Iris-setosa

Point 17: 5.4 3.9 1.3 0.4 - Iris-setosa

Point 18: 5.1 3.5 1.4 0.3 - Iris-setosa

Point 19: 5.7 3.8 1.7 0.3 - Iris-setosa

Point 20: 5.1 3.8 1.5 0.3 - Iris-setosa

Point 21: 5.4 3.4 1.7 0.2 - Iris-setosa

Point 22: 5.1 3.7 1.5 0.4 - Iris-setosa

Point 23: 4.6 3.6 1 0.2 - Iris-setosa

Point 24: 5.1 3.3 1.7 0.5 - Iris-setosa

Point 25: 4.8 3.4 1.9 0.2 - Iris-setosa

Point 26: 5 3 1.6 0.2 - Iris-setosa

Point 27: 5 3.4 1.6 0.4 - Iris-setosa

Point 28: 5.2 3.5 1.5 0.2 - Iris-setosa

Point 29: 5.2 3.4 1.4 0.2 - Iris-setosa

Point 30: 4.7 3.2 1.6 0.2 - Iris-setosa

Point 31: 4.8 3.1 1.6 0.2 - Iris-setosa

Point 32: 5.4 3.4 1.5 0.4 - Iris-setosa

Point 33: 5.2 4.1 1.5 0.1 - Iris-setosa

Point 34: 5.5 4.2 1.4 0.2 - Iris-setosa

Point 35: 4.9 3.1 1.5 0.1 - Iris-setosa

Point 36: 5 3.2 1.2 0.2 - Iris-setosa

Point 37: 5.5 3.5 1.3 0.2 - Iris-setosa

Point 38: 4.9 3.1 1.5 0.1 - Iris-setosa

Point 39: 4.4 3 1.3 0.2 - Iris-setosa

Point 40: 5.1 3.4 1.5 0.2 - Iris-setosa

Point 41: 5 3.5 1.3 0.3 - Iris-setosa

Point 42: 4.5 2.3 1.3 0.3 - Iris-setosa

Point 43: 4.4 3.2 1.3 0.2 - Iris-setosa

Point 44: 5 3.5 1.6 0.6 - Iris-setosa

Point 46: 4.8 3 1.4 0.3 - Iris-setosa

Point 47: 5.1 3.8 1.6 0.2 - Iris-setosa

Point 48: 4.6 3.2 1.4 0.2 - Iris-setosa

Point 49: 5.3 3.7 1.5 0.2 - Iris-setosa

Point 50: 5 3.3 1.4 0.2 - Iris-setosa

Cluster values: 5.006 3.418 1.464 0.244

Cluster 3

Point 64: 6.1 2.9 4.7 1.4 - Iris-versicolor

Point 51: 7 3.2 4.7 1.4 - Iris-versicolor

Point 52: 6.4 3.2 4.5 1.5 - Iris-versicolor

Point 54: 5.5 2.3 4 1.3 - Iris-versicolor

Point 55: 6.5 2.8 4.6 1.5 - Iris-versicolor

Point 56: 5.7 2.8 4.5 1.3 - Iris-versicolor

Point 57: 6.3 3.3 4.7 1.6 - Iris-versicolor

Point 58: 4.9 2.4 3.3 1 - Iris-versicolor

Point 59: 6.6 2.9 4.6 1.3 - Iris-versicolor

Point 60: 5.2 2.7 3.9 1.4 - Iris-versicolor

Point 61: 5 2 3.5 1 - Iris-versicolor

Point 62: 5.9 3 4.2 1.5 - Iris-versicolor

Point 63: 6 2.2 4 1 - Iris-versicolor

Point 65: 5.6 2.9 3.6 1.3 - Iris-versicolor

Point 66: 6.7 3.1 4.4 1.4 - Iris-versicolor

Point 67: 5.6 3 4.5 1.5 - Iris-versicolor

Point 68: 5.8 2.7 4.1 1 - Iris-versicolor

Point 69: 6.2 2.2 4.5 1.5 - Iris-versicolor

Point 70: 5.6 2.5 3.9 1.1 - Iris-versicolor

Point 71: 5.9 3.2 4.8 1.8 - Iris-versicolor

Point 72: 6.1 2.8 4 1.3 - Iris-versicolor

Point 73: 6.3 2.5 4.9 1.5 - Iris-versicolor

Point 74: 6.1 2.8 4.7 1.2 - Iris-versicolor

Point 75: 6.4 2.9 4.3 1.3 - Iris-versicolor

Point 76: 6.6 3 4.4 1.4 - Iris-versicolor

Point 77: 6.8 2.8 4.8 1.4 - Iris-versicolor

Point 79: 6 2.9 4.5 1.5 - Iris-versicolor

Point 80: 5.7 2.6 3.5 1 - Iris-versicolor

Point 81: 5.5 2.4 3.8 1.1 - Iris-versicolor

Point 82: 5.5 2.4 3.7 1 - Iris-versicolor

Point 83: 5.8 2.7 3.9 1.2 - Iris-versicolor

Point 84: 6 2.7 5.1 1.6 - Iris-versicolor

Point 85: 5.4 3 4.5 1.5 - Iris-versicolor

Point 86: 6 3.4 4.5 1.6 - Iris-versicolor

Point 87: 6.7 3.1 4.7 1.5 - Iris-versicolor

Point 88: 6.3 2.3 4.4 1.3 - Iris-versicolor

Point 89: 5.6 3 4.1 1.3 - Iris-versicolor

Point 90: 5.5 2.5 4 1.3 - Iris-versicolor

Point 91: 5.5 2.6 4.4 1.2 - Iris-versicolor

Point 92: 6.1 3 4.6 1.4 - Iris-versicolor

Point 93: 5.8 2.6 4 1.2 - Iris-versicolor

Point 94: 5 2.3 3.3 1 - Iris-versicolor

Point 95: 5.6 2.7 4.2 1.3 - Iris-versicolor

Point 96: 5.7 3 4.2 1.2 - Iris-versicolor

Point 97: 5.7 2.9 4.2 1.3 - Iris-versicolor

Point 98: 6.2 2.9 4.3 1.3 - Iris-versicolor

Point 100: 5.7 2.8 4.1 1.3 - Iris-versicolor

Point 102: 5.8 2.7 5.1 1.9 - Iris-virginica

Point 107: 4.9 2.5 4.5 1.7 - Iris-virginica

Point 114: 5.7 2.5 5 2 - Iris-virginica

Point 120: 6 2.2 5 1.5 - Iris-virginica

Point 122: 5.6 2.8 4.9 2 - Iris-virginica

Point 124: 6.3 2.7 4.9 1.8 - Iris-virginica

Point 127: 6.2 2.8 4.8 1.8 - Iris-virginica

Point 128: 6.1 3 4.9 1.8 - Iris-virginica

Point 134: 6.3 2.8 5.1 1.5 - Iris-virginica

Point 139: 6 3 4.8 1.8 - Iris-virginica

Point 143: 5.8 2.7 5.1 1.9 - Iris-virginica

Point 147: 6.3 2.5 5 1.9 - Iris-virginica

Point 150: 5.9 3 5.1 1.8 - Iris-virginica

Point 99: 5.1 2.5 3 1.1 - Iris-versicolor

Point 115: 5.8 2.8 5.1 2.4 - Iris-virginica

Cluster values: 5.90161 2.74839 4.39355 1.43387

[pict@localhost ~]$\*/