Ishika Prasad
ip1262@rit.edu (mailto:ip1262@rit.edu)

In [25]:
```
import random
import math
```

In [26]:
```
def calc_euclidist(d1, d2):
    distance = 0.0
    for i in range(0, len(d1)):
        distance += (d1[i] - d2[i])**2
    return math.sqrt(distance)
```

In [27]:
```
def calc_kmeans(points, k):
    clusters = []
    x = 0
    while x != k:
        clusters += [random.choice(points)]
        x += 1

    i = 0
    new_cluster = len(points) * [0]
    last_cluster = len(points) * [-1]
    while (new_cluster != last_cluster) or (i > 1000):
        last_cluster = list(new_cluster)
        i += 1

        for y in range(0, len(points)):
            minimum_dist = float("inf")
            for z in range(0, len(clusters)):
                distance = calc_euclidist(points[y], clusters[z])
                if distance < minimum_dist:
                    minimum_dist = distance
                    new_cluster[y] = z

        for a in range(0, len(clusters)):
            new_centroid = len(points[0]) * [0]
            mem = 0
            for b in range(0, len(points)):
                if new_cluster[b] == a:
                    for c in range(0, len(points[0])):
                        new_centroid[c] += points[b][c]
                    mem += 1
            for m in range(0, len(points[0])):
                if mem != 0:
                    new_centroid[m] = new_centroid[m] / float(mem)
            clusters[a] = new_centroid

    print("Clusters : ", clusters)
```

In [36]:

```
points = [(3, 2), (2, 2), (1, 2), (1, 1), (5, 6), (7, 7), (9, 10), (11, 13),
k = 4
calc_kmeans(points, k)
```

```
Clusters :  [[2.0, 2.0], [11.666666666666666, 12.666666666666666], [1.0, 1.
0], [7.0, 7.666666666666667]]
```