




알고리즘 실습

1. 클러스터링



- NP(Nondeterministic Polynomial) Problem
 - ✓ 다항 시간 내에 검증될 수 있는 해가 주어진 경우, 그 해를 다항 시간 내에 찾을 수 있다는 의미
- Polynomial-time reduction
 - ✓ 문제 A와 B가 있을 때, A의 해를 다항 시간 안에 B의 해로 변환할 수 있는 것
- 문제 X가 NP와 Polynomial time reduction을 만족하면 X는 NP-Complete
- 현재까지 NP-Complete 문제에 대해 다항 시간 내에 풀 수 있는 알고리즘은 개발되지 않음

➤ Clustering 문제를 최적으로 해결하기 위해서는

✓ 모든 가능한 클러스터링 구성을 확인해야 함(1)

탐색에 지수 시간이 걸림

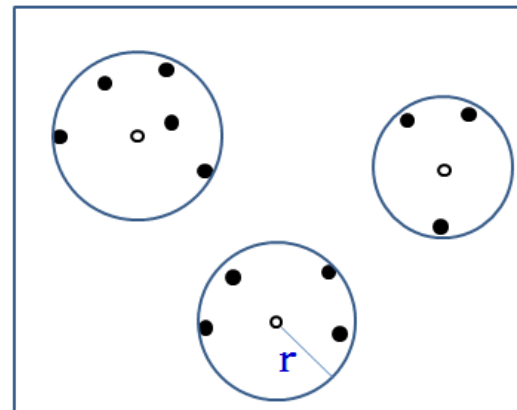
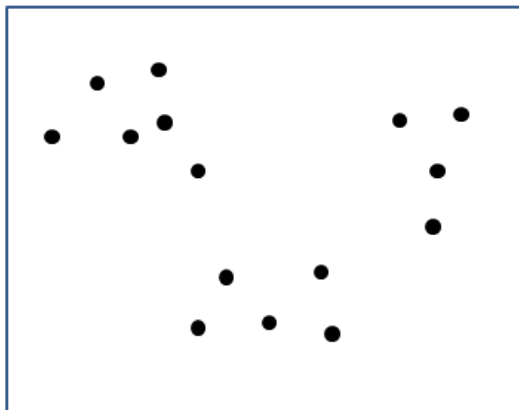
✓ 각 구성에 대해 최소 비용을 가지는 것을 찾아야함(2)

➤ 따라서 근사 알고리즘은 최적해 대신 다항 시간 안에 정해진 근사 비율 내에서 해를 찾도록 설계됨

➤ Clustering 알고리즘의 근사 비율은 2.0인데 이는 Clustering의 근사해가 최적해의 최대 2배를 넘지 않음을 의미

8.5 클러스터링 문제

- 2차원 평면의 n 개의 점들 간의 거리를 고려하여 k 개의 그룹으로 나누자.

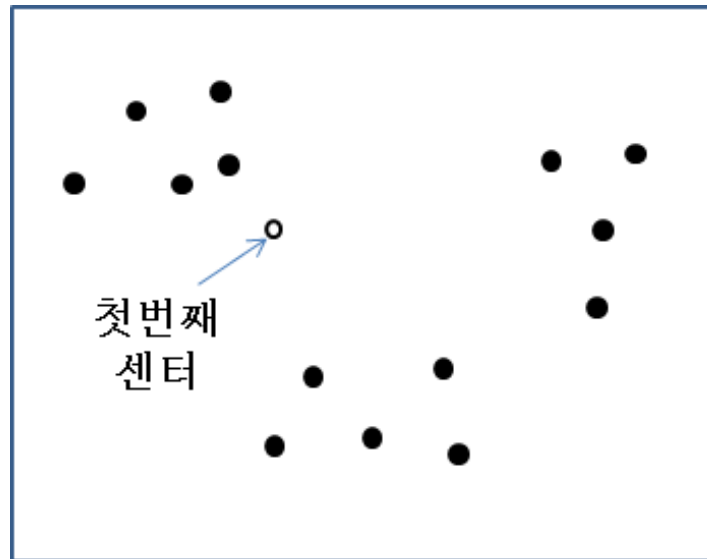


- 클러스터링 (Clustering) 문제

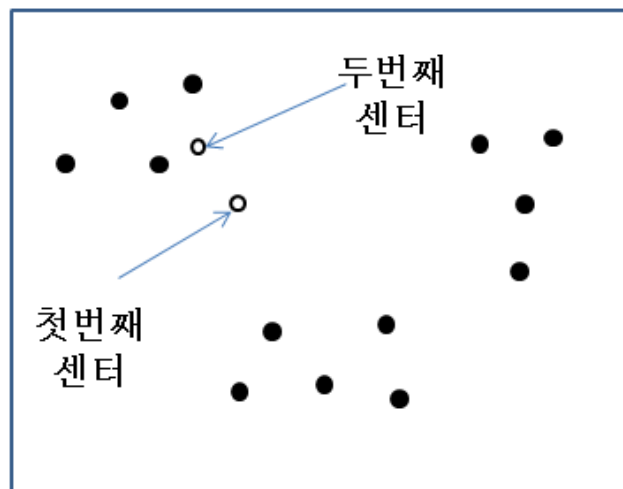
- n 개의 점을 k 개의 그룹으로 나누고 각 그룹의 중심이 되는 k 개의 점을 선택하는 문제
- 단, 가장 큰 반경을 가진 그룹의 직경이 최소가 되도록 k 개의 점을 선택해야함.

❖ k개의 센터 선택 방법

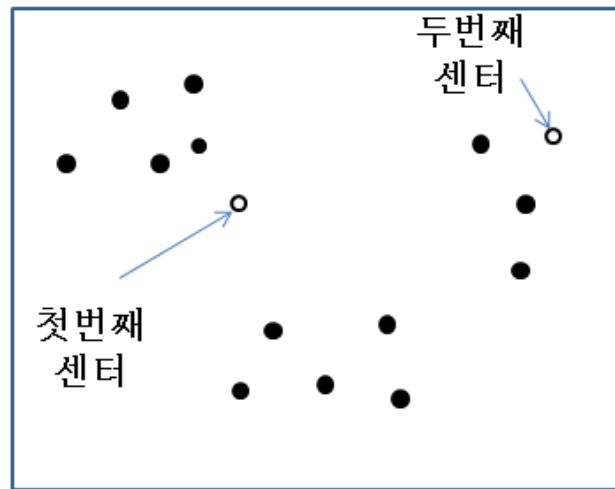
- 1개씩 선택
- 임의의 랜덤한 점을 첫 번째 센터로 선택



➤ 두 번째 센터는 어느 점이 좋을까?



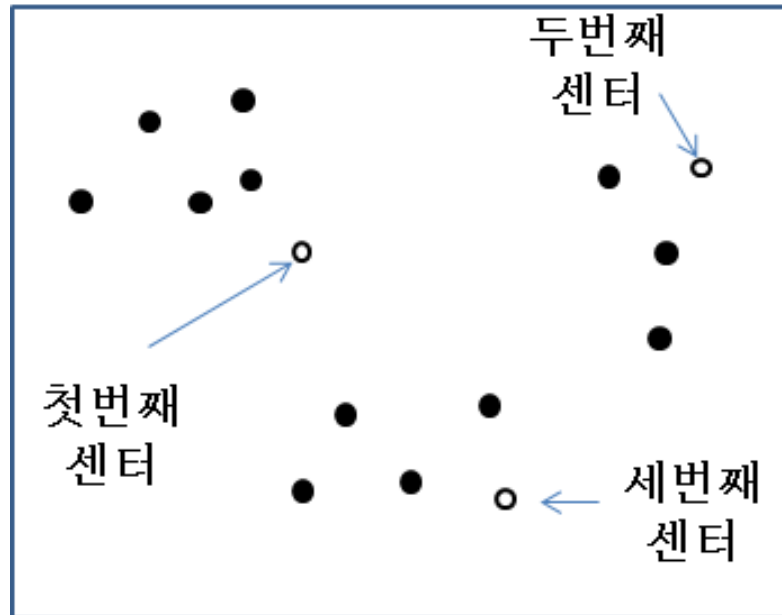
첫번째 센터에서 가장 가까운 점



첫번째 센터에서 가장 먼 점

➤ 두 개의 센터가 서로 가까이 있는 것보다 멀리 떨어져 있는 것이 좋음.

➤ 세 번째 센터는?



➤ 첫 번째와 두 번째 센터 둘 다에서 가장 멀리 떨어진 점을 선택

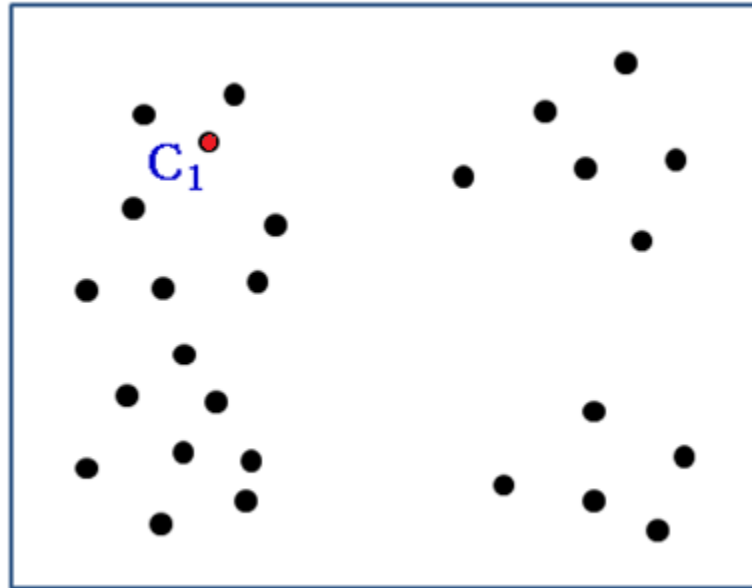
입력: n 개의 점 $x_i, i=0, 1, \dots, n-1$, 그룹의 수 $k > 1$

출력: k 개의 점의 그룹 및 각 그룹의 센터

1. $C[1] = r$, 단, x_r 은 랜덤하게 선택
2. **for** $j = 2$ to k
3. **for** $i = 0$ to $n-1$
4. **if** $x_i \neq \text{센터}$
5. x_i 와 각 센터까지의 거리를 계산하여, x_i 와 가장 가까운
 센터까지의 거리를 $D[i]$ 에 저장한다.
6. $C[j] = i$, 단, i 는 D 의 가장 큰 원소의 인덱스이고, x_i 는 센터가 아니다.
7. 센터가 아닌 각 점 x_i 로부터 앞서 찾은 k 개의 센터까지 거리를 각각 계산하고 그
 중에서 가장 짧은 거리의 센터를 찾는다. 이때 점 x_i 는 가장 가까운 센터의 그룹에
 속하게 된다.
8. **return** // C 와 각 클러스터에 속한 점들의 리스트

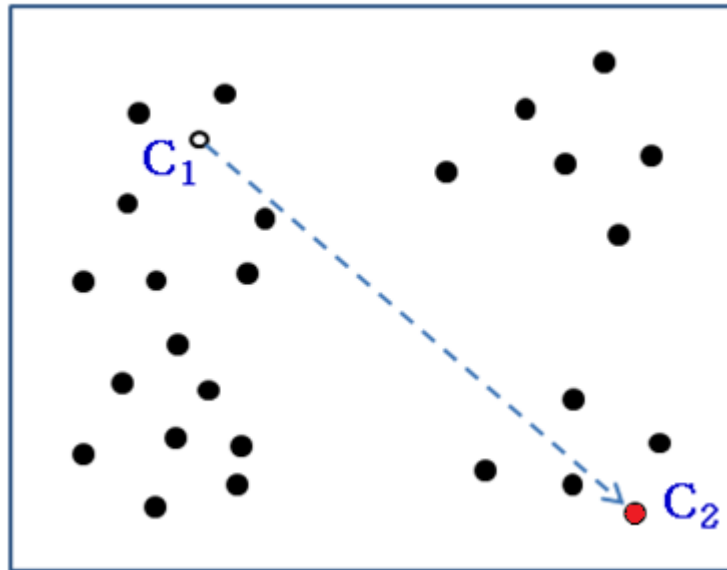
❖ 첫 번째 센터

- 임의의 점 하나를 첫 번째 센터 C_1 으로 ($k=4$)



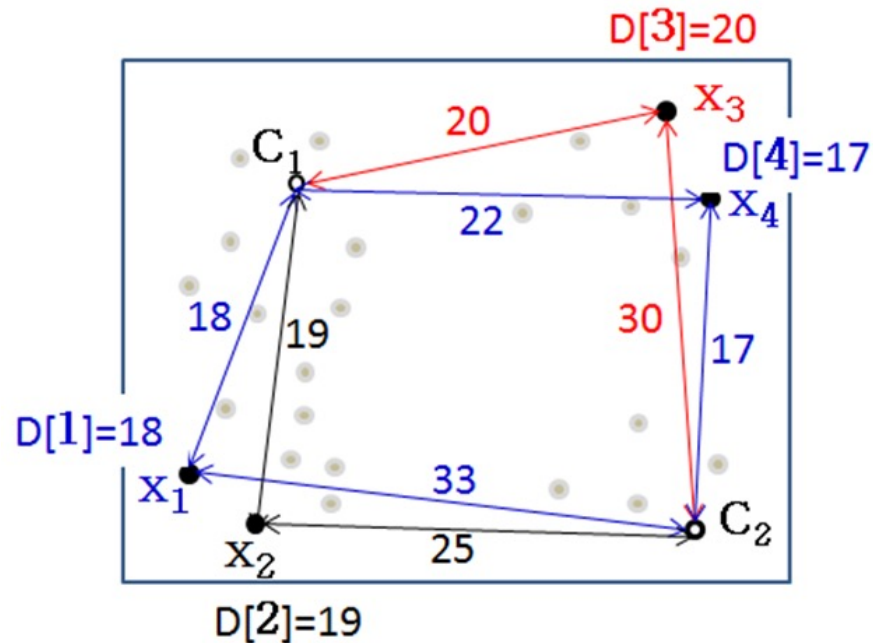
❖ 두 번째 센터

- C_1 이 아닌 각 점 x_i 에서 C_1 까지의 거리 $D[i]$ 계산
- C_2 로부터 거리가 가장 먼 점을 다음 센터 C_2 로

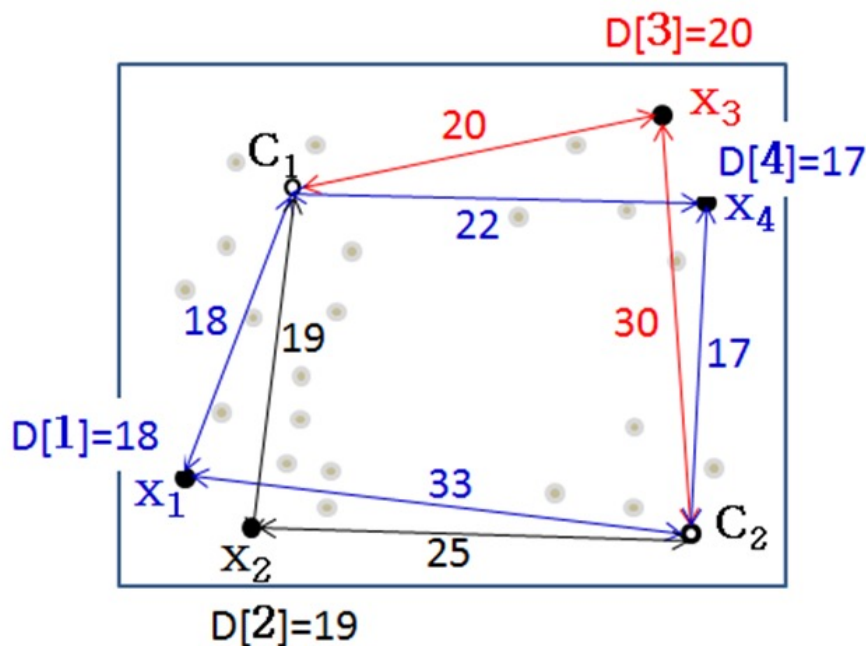


❖ 세 번째 센터

- C_1 과 C_2 를 제외한 각 점 x_i 에서 각각 C_1 과 C_2 까지의 거리를 계산하여 그 중에서 작은 값을 $D[i]$ 로 정한다.
- D 에서 가장 큰 값을 가진 원소의 인덱스가 i 라고 하면, 점 x_i 가 C_3 이 된다.



❖ $D[i]$ 계산



$D[1] = 18, \min\{\text{dist}(x_1, C_1), \text{dist}(x_1, C_2)\}$
 $= \min\{18, 33\}$ 이므로

$D[2] = 19, \min\{\text{dist}(x_2, C_1), \text{dist}(x_2, C_2)\}$
 $= \min\{19, 25\}$ 이므로

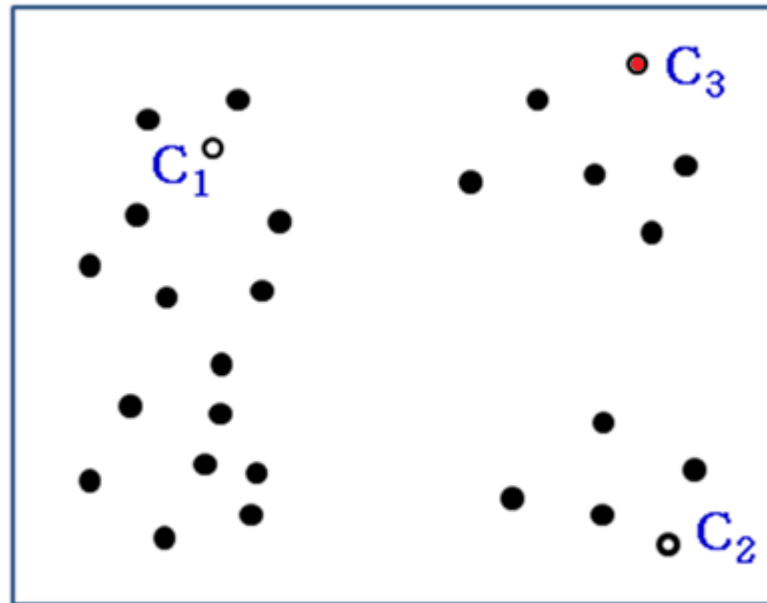
$D[3] = 20$, $\min\{\text{dist}(x_3, C_1), \text{dist}(x_3, C_2)\}$
 $= \min\{20, 30\}$ 이므로

$D[4] = 17, \min\{\text{dist}(x_4, C_1), \text{dist}(x_4, C_2)\}$
 $= \min\{22, 17\}$ 이므로

다른 x_i 의 $D[i]$ 는 20보다 작다고 가정

➤ $\text{dist}(x, C)$ 는 점 x 와 센터 C 사이의 거리

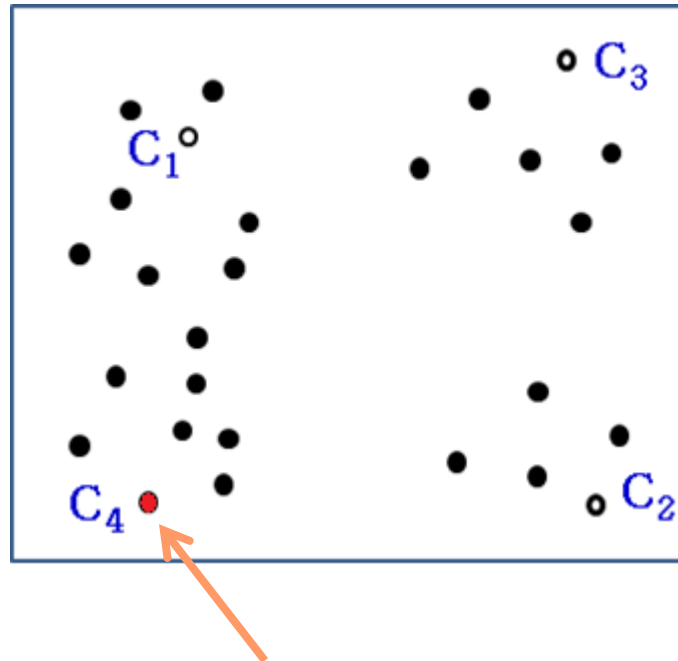
➤ $D[3]$ 이 가장 큰 값이므로



C_1 과 C_2 로부터 가장 먼 점

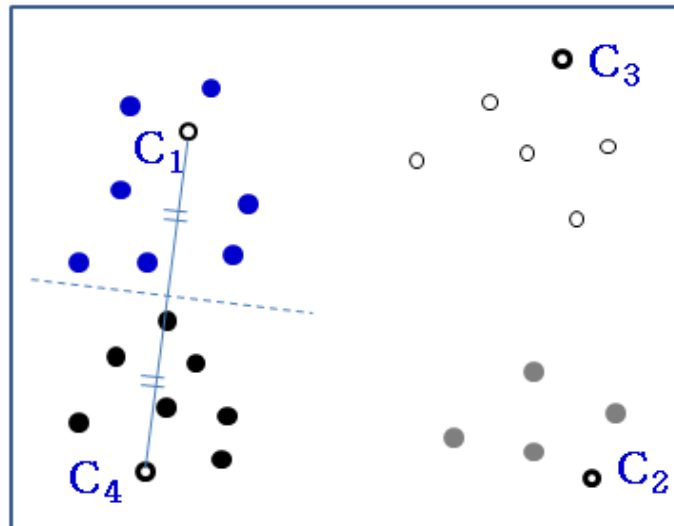
❖ 네 번째 센터

- C_1, C_2, C_3 을 제외한 각 점 x_i 에서 각각 C_1, C_2, C_3 까지의 거리를 계산하여 그 중에서 작은 값을 $D[i]$ 로 정한다.
- D 에서 가장 큰 값을 가진 원소의 인덱스가 i 이면, 점 x_i 가 C_4



❖ 그룹으로 나누기

- 센터가 아닌 각 점 x_i 로부터 위에서 찾은 4개의 센터까지 거리를 각각 계산하고 그 중에 가장 짧은 거리의 센터를 찾는다.
- x_i 는 가장 가까운 센터의 그룹에 속하게 된다.



❖ 시간 복잡도

- 내부 for-루프: 각 점에서 각 센터까지의 거리를 계산하므로 $O(kn)$ 시간
- Line 6 최대값을 찾으므로 $O(n)$ 시간
- 외부 for-루프는 $(k - 1)$ 회 반복하므로

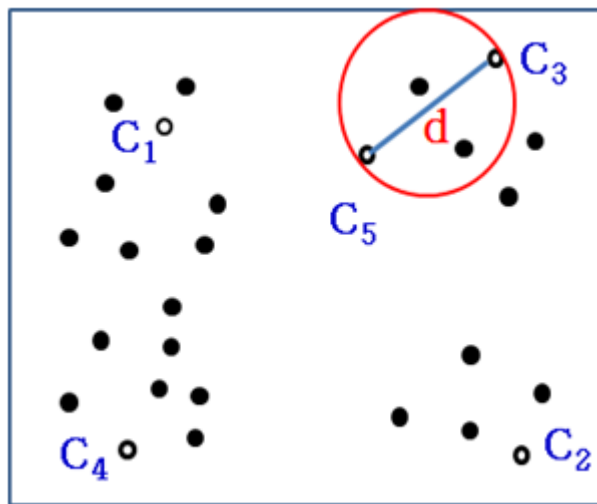
$$O(1) + (k - 1) \times (O(kn) + O(n))$$

- Line 7 센터가 아닌 각 점으로부터 k 개의 센터까지의 거리를 각각 계산하면서 최솟값을 찾으므로 $O(kn)$ 시간

$$O(1) + (k - 1) \times (O(kn) + O(n)) + O(kn) = O(k^2n)$$

❖ 근사 비율

- 최적해가 만든 그룹 중에서 가장 큰 직경을 OPT
- OPT 의 하한을 간접적으로 찾기 위해 알고리즘이 k 개의 센터를 모두 찾고 나서 $(k + 1)$ 번째 센터를 찾은 상황에서, 즉, $k = 4$ 일 때, 1개의 센터 C_5 를 추가한 상황을 살펴보자.

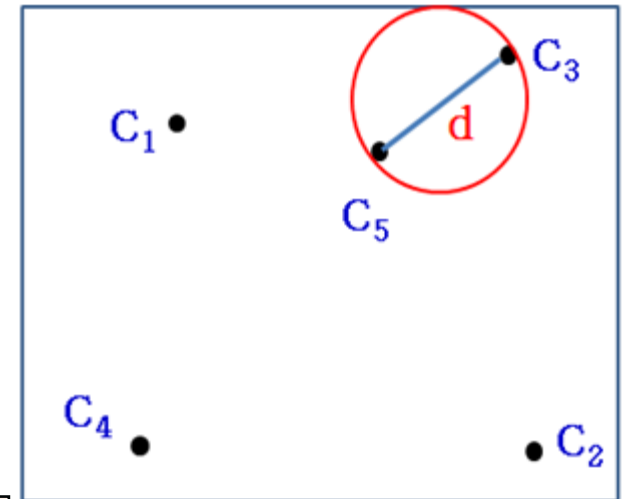


❖ 근사 비율

- C_5 에서 가장 가까운 센터인 C_3 까지의 거리를 d 라고 하면
- 클러스터링 문제의 최적해를 계산하는 어떤 알고리즘이라도 위의 5개의 센터 점을 ($k = 4$ 이니까) 4개의 그룹으로 분할해야 한다.
- 따라서 5개의 센터 중에서 2개는 하나의 그룹에 속해야만 한다.
- 그림에서는 C_3 과 C_5 가 하나의 그룹에 속한다.
- 최적해의 가장 큰 그룹의 직경인

OPT 는 d 보다 작을 수는 없다. 즉, $OPT \geq d$ 이다.

- $OPT \geq d$ 이고, $OPT' \leq 2d$ 이므로,
- $2OPT \geq 2d \geq OPT'$
- Approx_k_Clusters 알고리즘의 근사 비율은 2.0



최적해보다 최대 2배 더
나쁠 수 있다는 이야기

- ❖ 첫 센터를 랜덤하게 선택하므로 보다 나은 클러스터링을 위해 알고리즘을 여러 차례 수행하여 얻은 결과 중에 best 클러스터링을 사용한다.
- ❖ 비정상적인 데이터(노이즈, outlier)에 취약한 성능을 보이므로 선처리를 통해 이들을 제거 후 사용해야 한다.

❖ 1030 문제를 푸시오

➤ Status와 ID가 보이는 캡처 사진과 소스코드 + 주석을 포함

Description

한 공간에 N명의 사람들이 각각의 좌표 x, y 에 있어야 한다. 사람들 사이에는 시민과 M명의 경찰이 포함되어 있다.

경찰과의 거리가 멀수록 시민은 불안함을 느낀다. 따라서, 각 시민의 불안도는 배치된 경찰 중 가장 가까운 경찰과의 거리와 같다.

Approximate k Clustering 알고리즘을 이용하여 가장 효율적으로 배치된 경찰의 좌표를 구하라.

(단, 좌표간 거리는 Euclidean distance로 구한다.)

Input

첫 번째 줄에는 사람의 수 N과 경찰의 수 M가 공백으로 구분되어 주어진다. ($1 \leq M \leq N \leq 1000$)

두 번째 줄부터 N개의 사람이 위치 가능한 좌표 x, y 가 공백으로 구분되어 주어진다. ($0 \leq x, y \leq 10000$)

여기서, 처음 주어진 좌표는 첫 번째 경찰이 배치될 좌표이다.

Output

시민이 느낄 위험 정도가 가능한 한 작아지도록 하는 공백으로 구분된 경찰 M명의 좌표 x, y 가 한 줄에 하나씩 M개 출력한다.