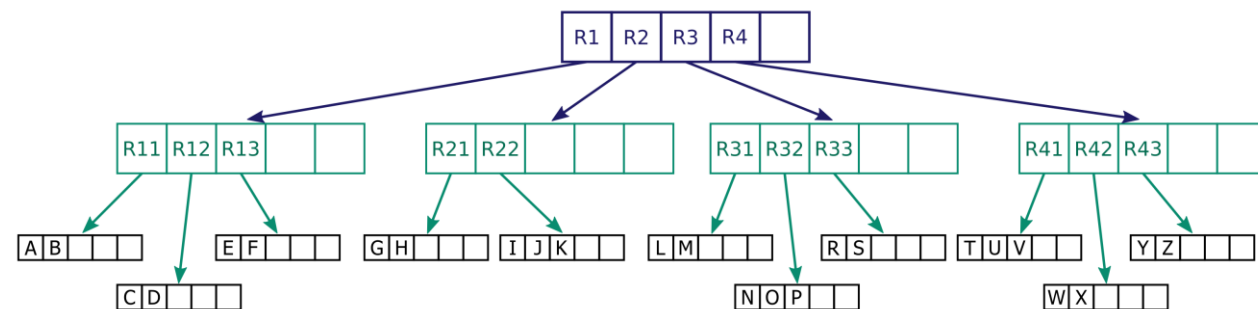
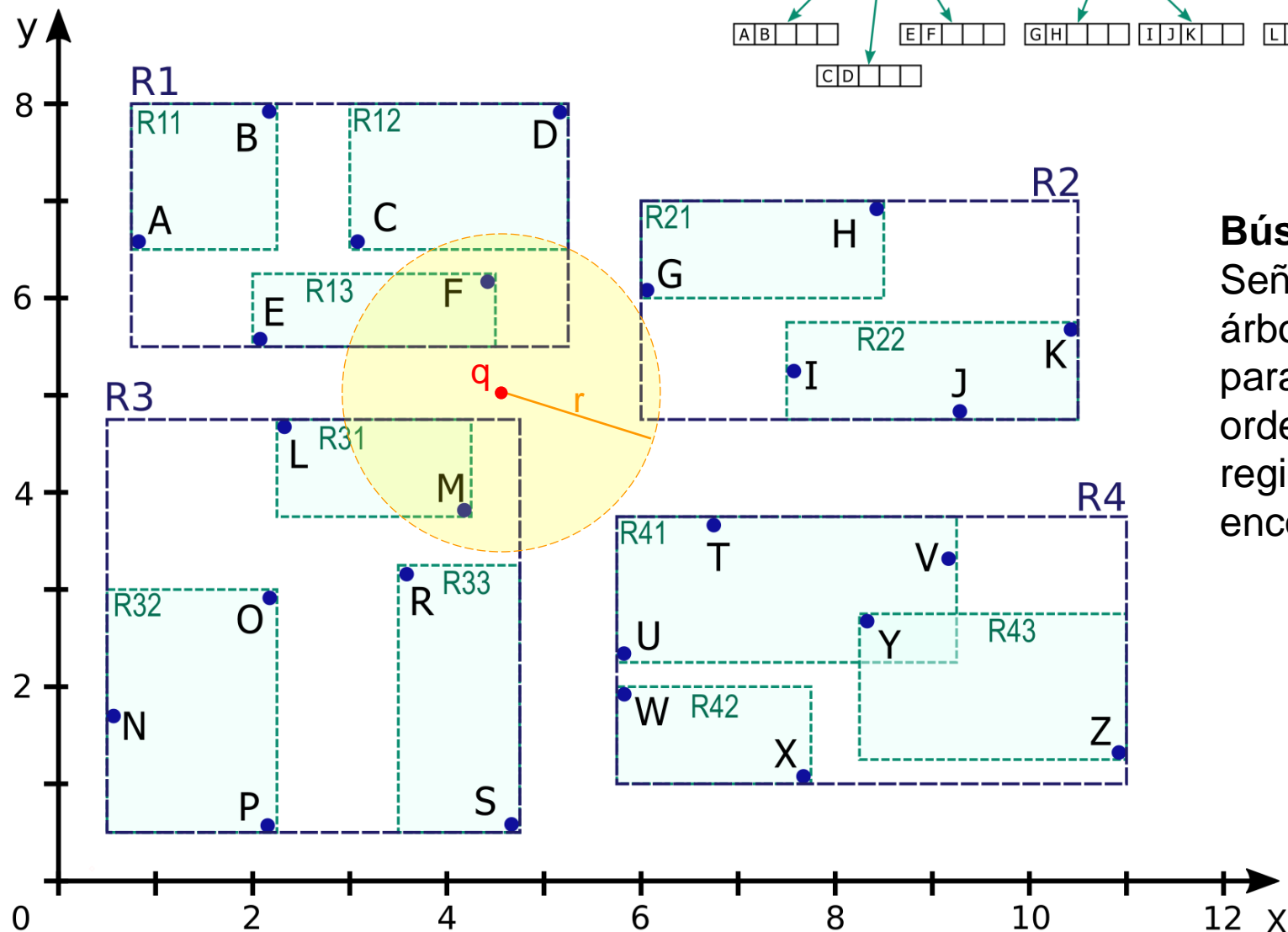


Ejercicio 1 R-tree



Búsqueda por rango (q,r):
Señalar todas las regiones del árbol que deben ser visitadas para resolver la búsqueda, el orden en que se visitan las regiones y los elementos encontrados.

Respuesta Ejercicio 1

Algoritmo Búsqueda por rango usando recursividad (recorrido izq-der)

- Región R1 (intersecta Bola de consulta, ingresa a la región)
 - Región R11 (no intersecta Bola de consulta, no ingresa)
 - Región R12 (sí intersecta Bola de consulta, ingresa)
 - Punto C (no es reportado)
 - Punto D (no es reportado)
 - Región R13 (sí intersecta Bola de consulta, ingresa)
 - Punto E (no es reportado)
 - Punto F (sí es reportado como resultado)
- Región R2 (sí intersecta Bola de consulta, ingresa)
 - Región R21 (no intersecta Bola de consulta, no ingresa)
 - Región R22 (no intersecta Bola de consulta, no ingresa)
- Región R3 (sí intersecta Bola de consulta, ingresa)
 - Región R31 (sí intersecta Bola de consulta, ingresa)
 - Punto L (no es reportado)
 - Punto M (sí es reportado como resultado)
 - Región R32 (no intersecta Bola de consulta, no ingresa)
 - Región R33 (no intersecta Bola de consulta, no ingresa)
- Región R4 (no intersecta Bola de consulta, no ingresa)
- Fin (se recorrió todo el árbol)

Resumen

Regiones visitadas:

- R1, R12, R13, R2, R3, R31

Elementos comparados:

- C, D, E, F, L, M

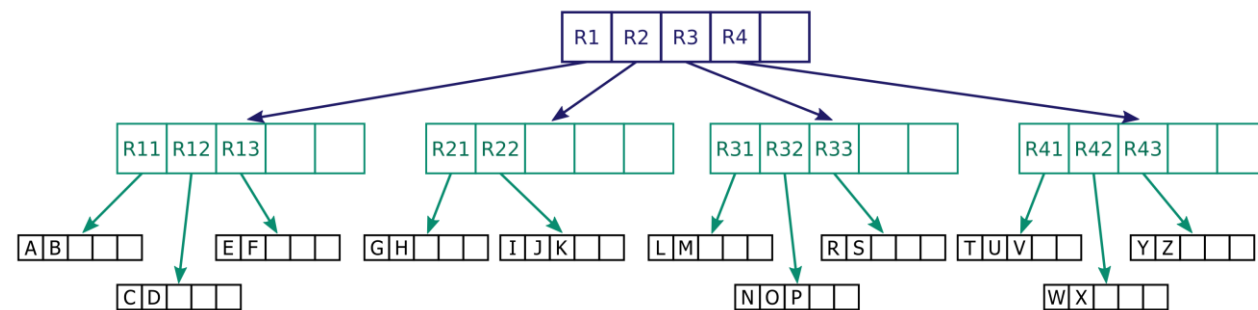
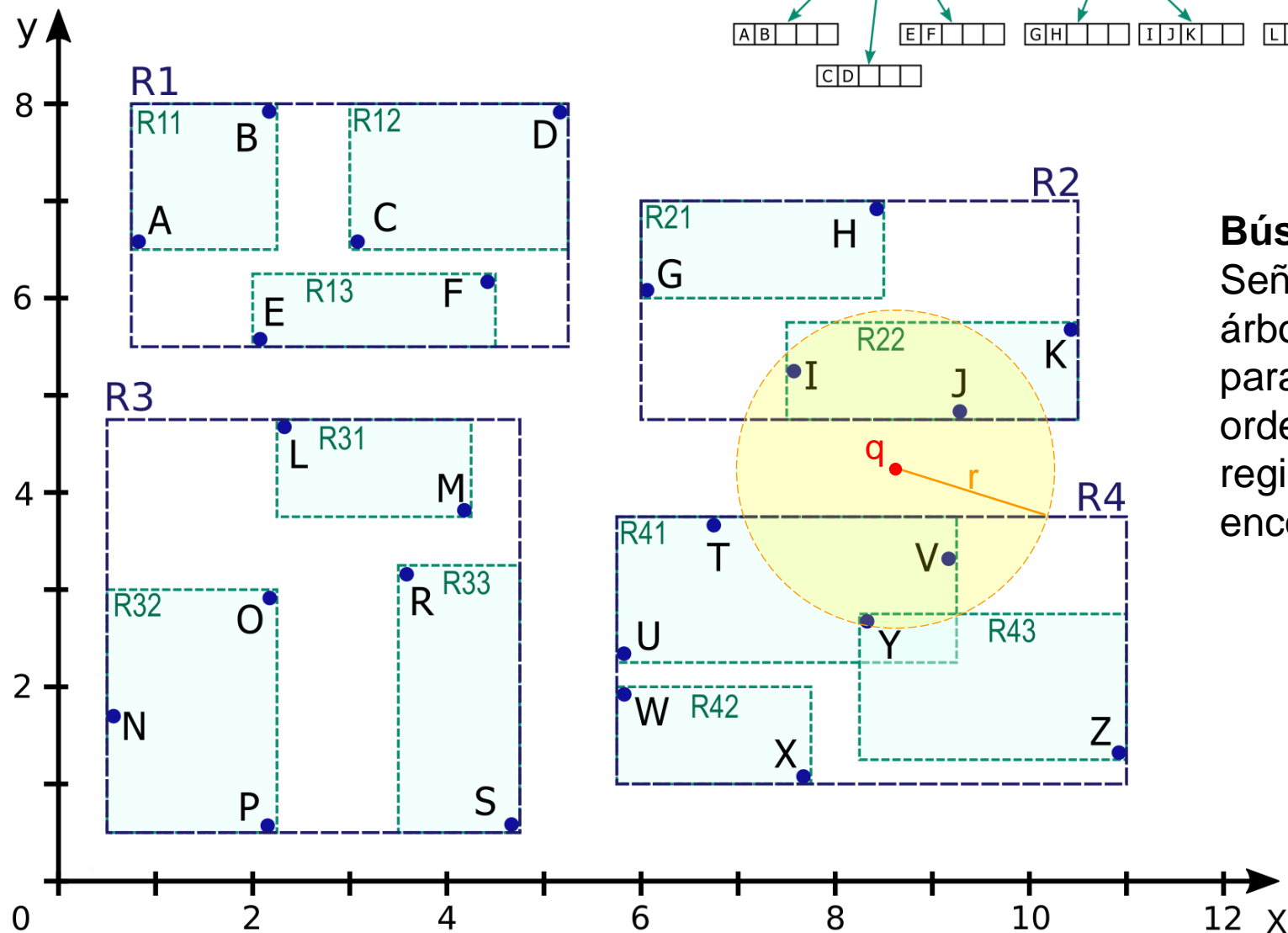
Elementos encontrados:

- F, M

Cálculos de distancia:

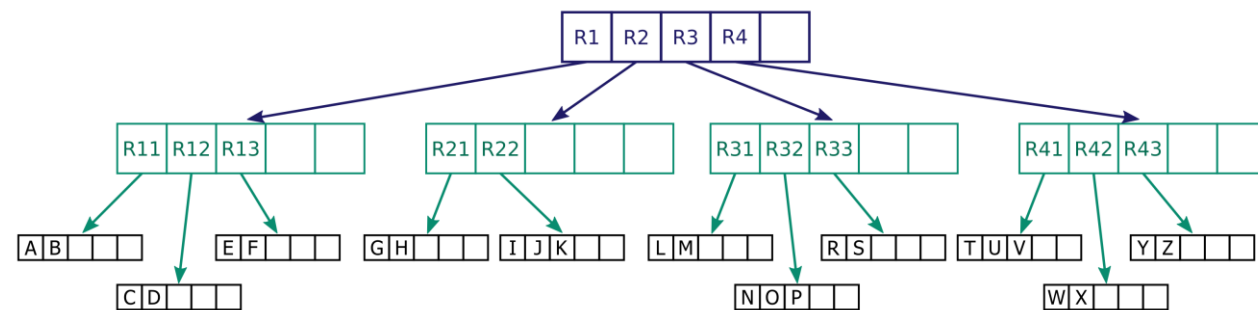
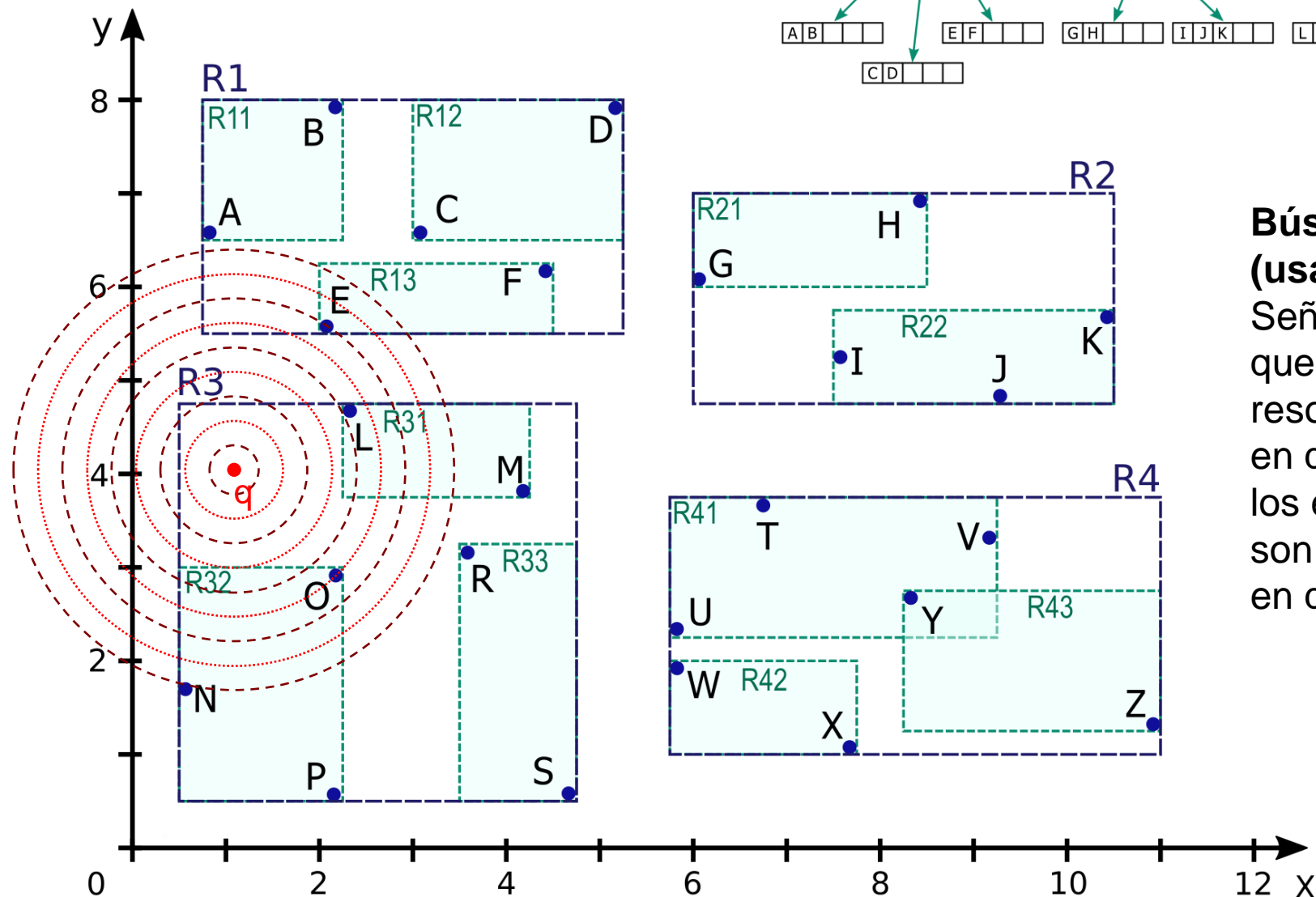
- 12 veces se evalúa una distancia entre q y una región
- 6 veces se evalúa una distancia entre q y un elemento

Ejercicio 2 R-tree



Búsqueda por rango (q,r):
Señalar todas las regiones del árbol que deben ser visitadas para resolver la búsqueda, el orden en que se visitan las regiones y los elementos encontrados.

Ejercicio 3 k-NN



Búsqueda del 1-nn de q (usando cola de prioridad): Señalar las regiones del árbol que deben ser visitadas para resolver la búsqueda, el orden en que se visitan las regiones, los elementos candidatos que son encontrados y el momento en que finaliza la búsqueda.

Respuesta Ejercicio 3

Algoritmo Búsqueda del NN usando una Cola de Prioridad.

La cola de prioridad permite obtener rápidamente la región de menor prioridad (min-heap). Se usa como valor de prioridad el MINDIST de la región a la consulta.

- Se inicializa cola de prioridad=(), candidato a NN=null, distancia de corte pruning_dist=infinito
- Se visita nodo raíz y se obtienen regiones R1, R2, R3 y R4
- Se calculan MINDIST de q a R1, R2, R3 y R4 y se agregan a la cola de prioridad (todas son menores que pruning_dist)
 - cola de prioridad=(R3, R1, R4, R2)
- Extrae la región de menor MINDIST en la cola: R3
- Visita R3 y obtiene regiones R31, R32, R33
- Se calculan MINDIST a q a R31, R32, R33 y se agregan a la cola de prioridad (todas son menores que pruning_dist)
 - cola de prioridad=(R31, R32, R1, R33, R4, R2)
- Extrae la región de menor MINDIST en la cola: R31
- Visita R31 y obtiene dos puntos
 - punto L (compara distancia, es el nuevo candidato a NN, se fija pruning_dist)
 - punto M (compara distancia, es más lejano que el candidato a NN, descartado)
- Extrae la región de menor MINDIST en la cola: R32
- Visita R32 y obtiene tres puntos
 - punto N (compara distancia, es más lejano que el candidato a NN, descartado)
 - punto O (compara distancia, es más lejano que el candidato a NN, descartado)
 - punto P (compara distancia, es más lejano que el candidato a NN, descartado)
- Extrae la región de menor MINDIST en la cola: R1
- Fin del algoritmo (el valor de MINDIST de R1 es mayor que pruning_dist)

Resumen

Regiones visitadas:

- R3, R31, R32

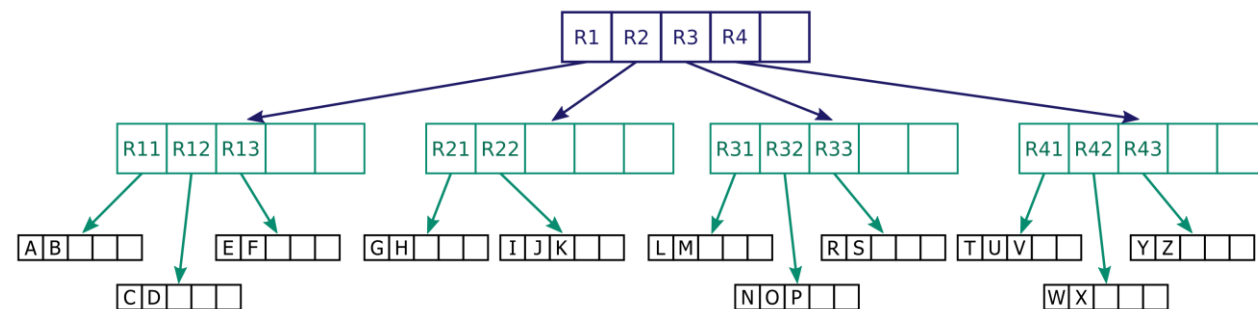
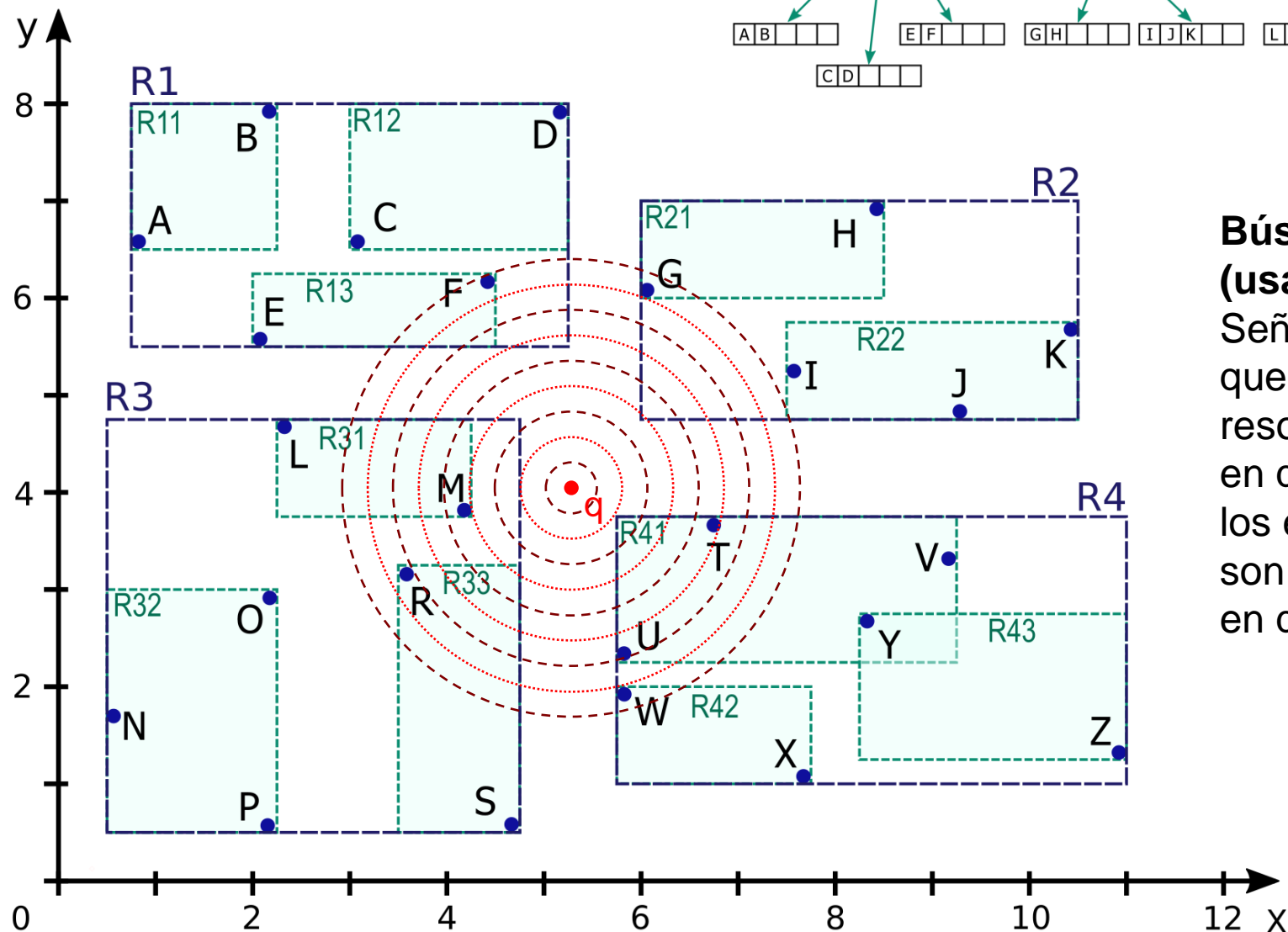
Elementos comparados:

- L, M, N, O, P

Cálculos de distancia:

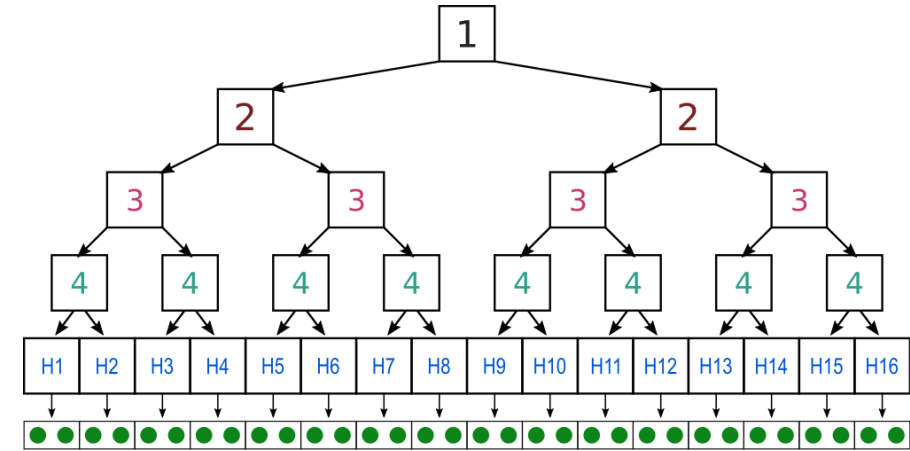
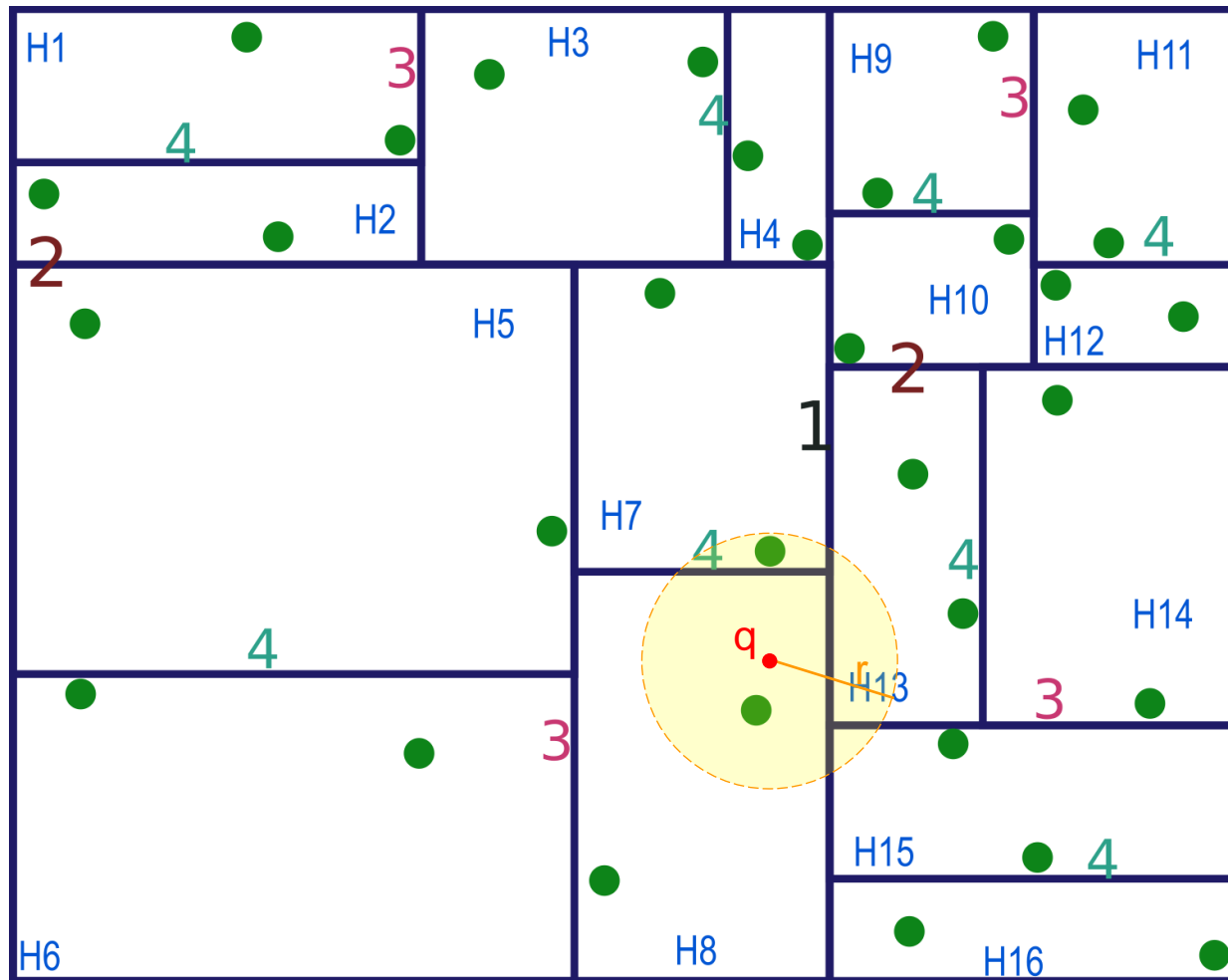
- 7 veces se calcula MINDIST entre q y una región
- 4 veces se calcula distancia entre q y un elemento

Ejercicio 4 k-NN



Búsqueda del 1-nn de q (usando cola de prioridad): Señalar las regiones del árbol que deben ser visitadas para resolver la búsqueda, el orden en que se visitan las regiones, los elementos candidatos que son encontrados y el momento en que finaliza la búsqueda.

Ejercicio 5 k-d tree

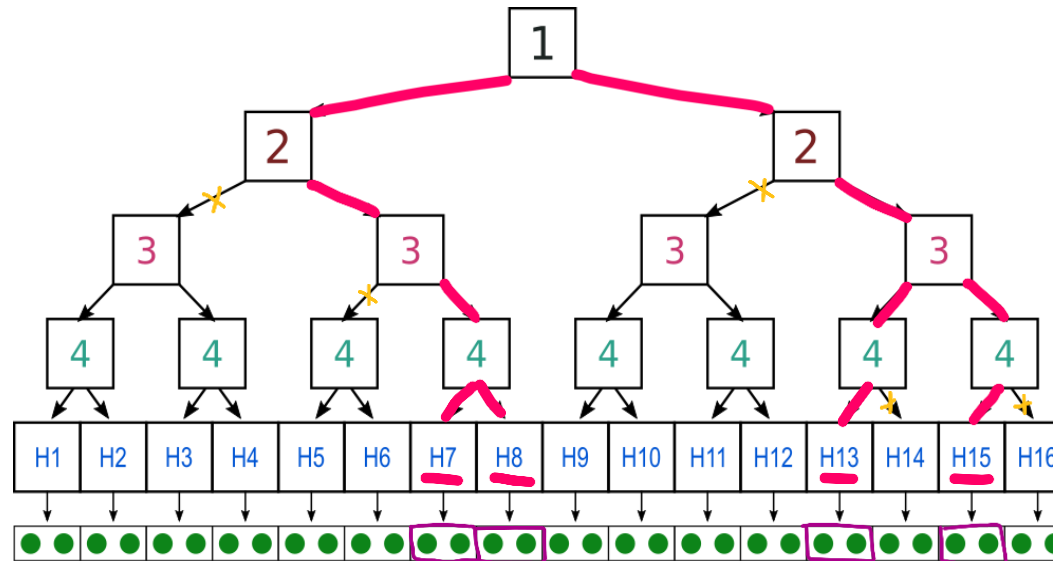


Búsqueda por rango (q,r):
Señalar todas las hojas del árbol que deben ser visitadas para resolver la búsqueda, el orden en que se visitan las regiones y los elementos encontrados.

Respuesta Ejercicio 5

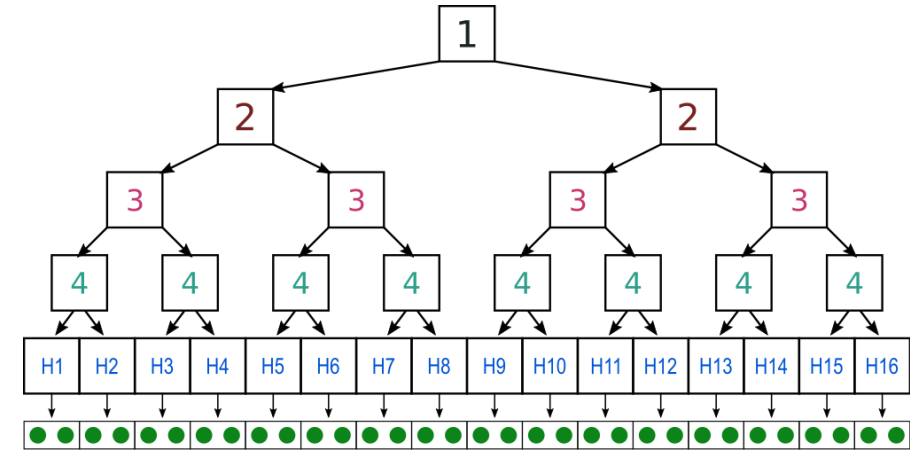
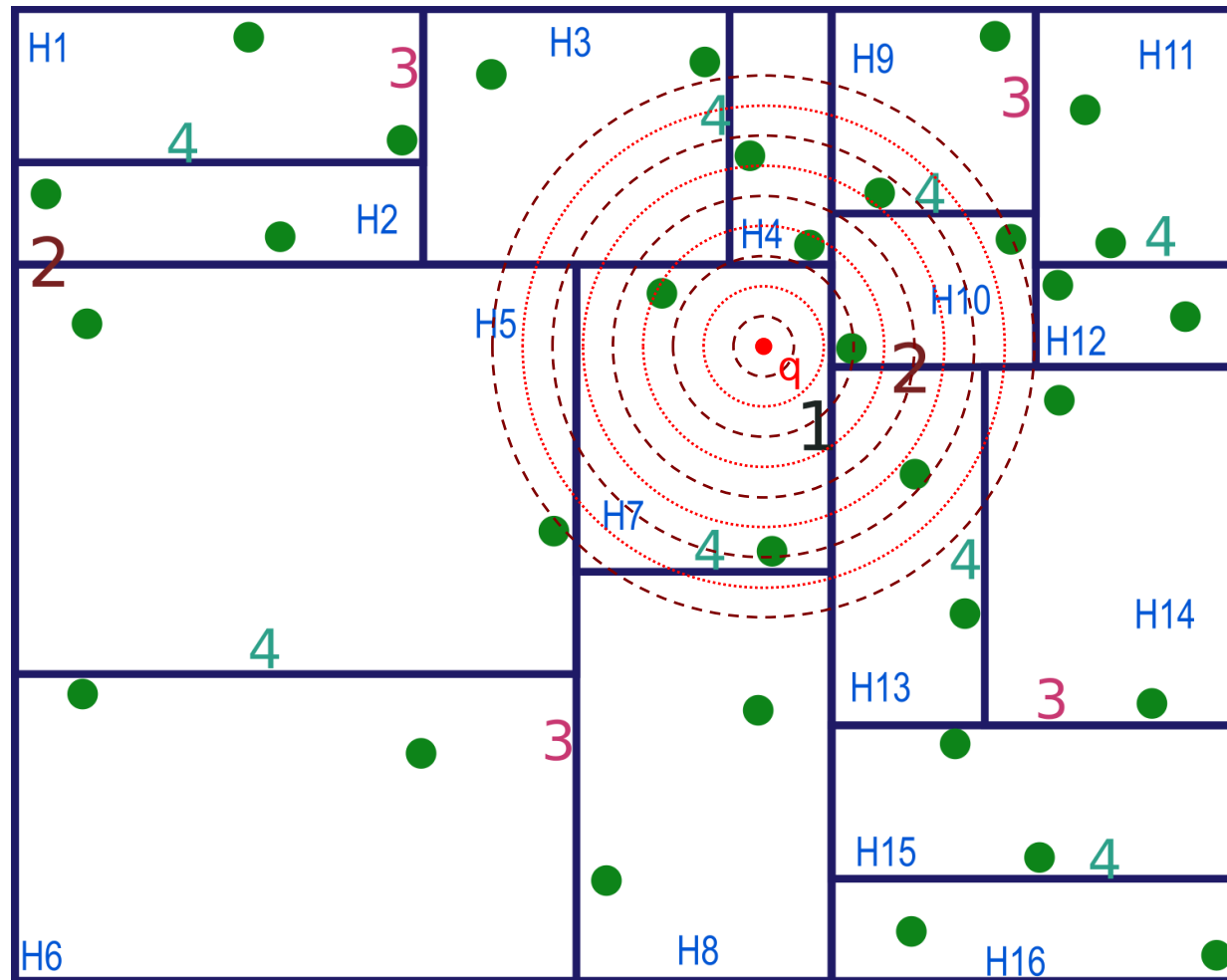
Búsqueda por rango usando recursividad (recorrido izq-der)

Recorre el árbol recursivamente y visita solo las hojas: H7, H8, H13 y H15 (en ese orden).



Calcula la distancia euclidiana de 8 elementos y reporta 2 elementos que están dentro de la bola de consulta (uno en H7 y uno en H8).

Ejercicio 6 k-d tree



Búsqueda del 1-nn de q (usando cola de prioridad):
Señalar las hojas del árbol que son visitadas para resolver la búsqueda 1-NN, el orden en que se visitan, los elementos candidatos que son encontrados y el momento en que finaliza la búsqueda.

Respuesta Ejercicio 6

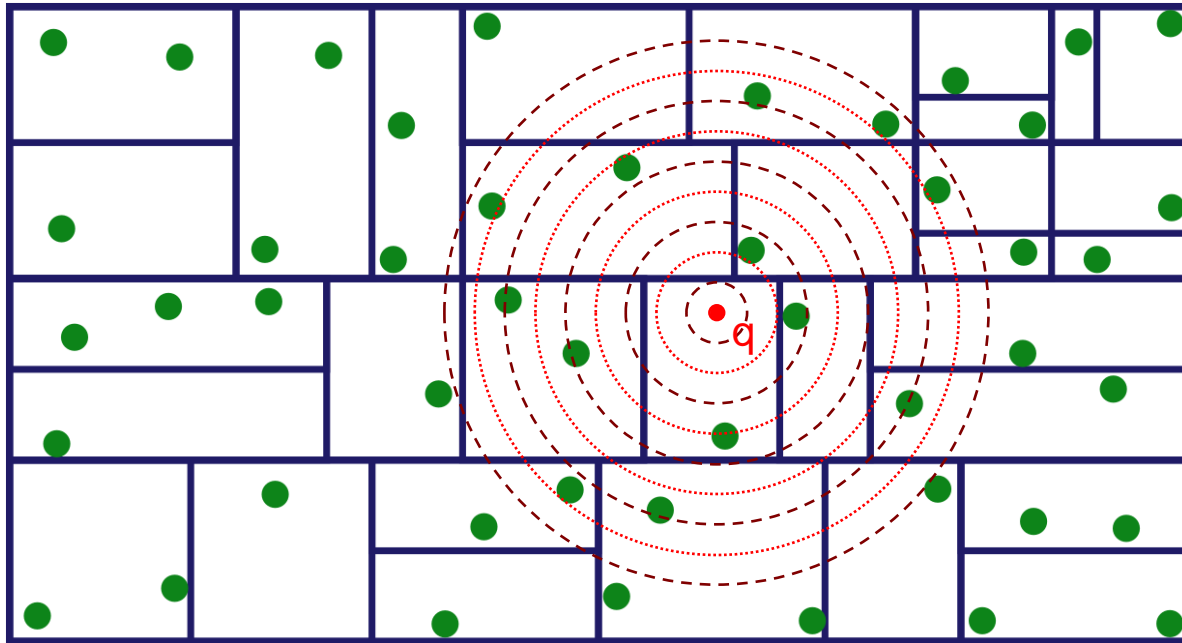
Búsqueda del NN usando una Cola de Prioridad en el k-d tree.

- La primera hoja visitada es H7, ya que tiene menor MINDIST (0)
 - Compara los dos elementos de la región y obtiene primer candidato.
 - Fija pruning_dist en el círculo #4.
- La segunda hoja visitada es H10
 - Compara los dos elementos de la región y encuentra un nuevo candidato.
 - Reduce pruning_dist al círculo #3.
- La tercera hoja visitada es H13
 - Compara los dos elementos de la región, pero no mejora pruning_dist
- La cuarta hoja visitada es H4
 - Compara los dos elementos de la región, pero no mejora pruning_dist
- Fin (la siguiente región es H3, pero su MINDIST es mayor que pruning_dist)

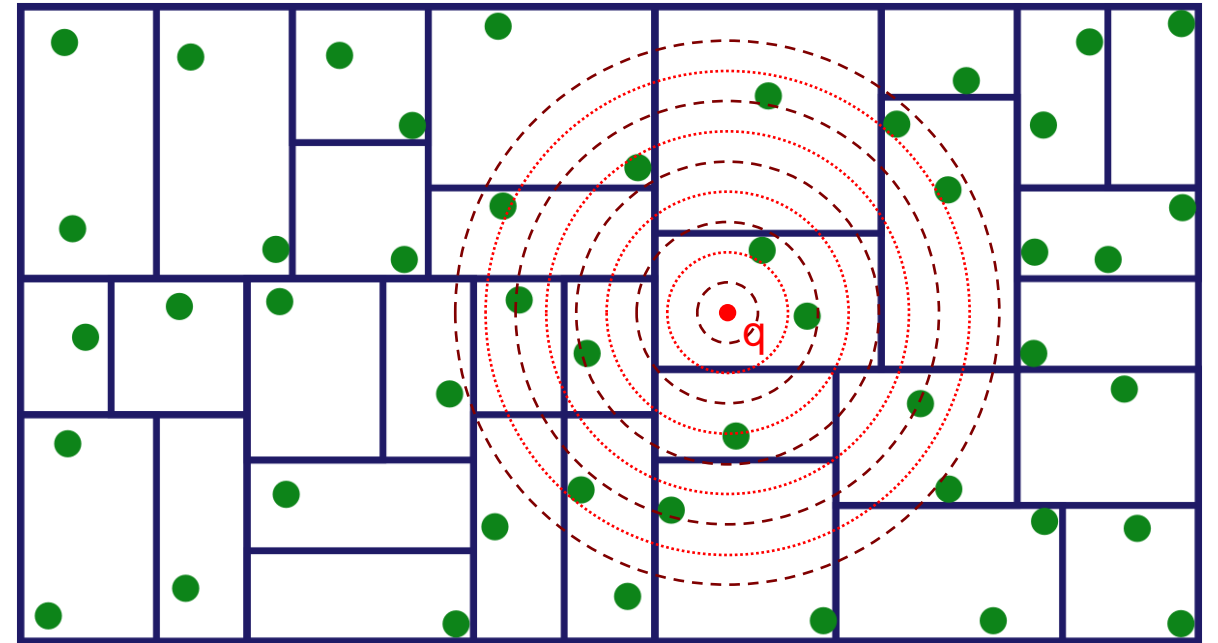
Ejercicio 7 Randomized k-d trees

(se muestran las regiones hoja de 2 randomized k-d tree creados sobre los mismos datos)

k-d tree 1



k-d tree 2

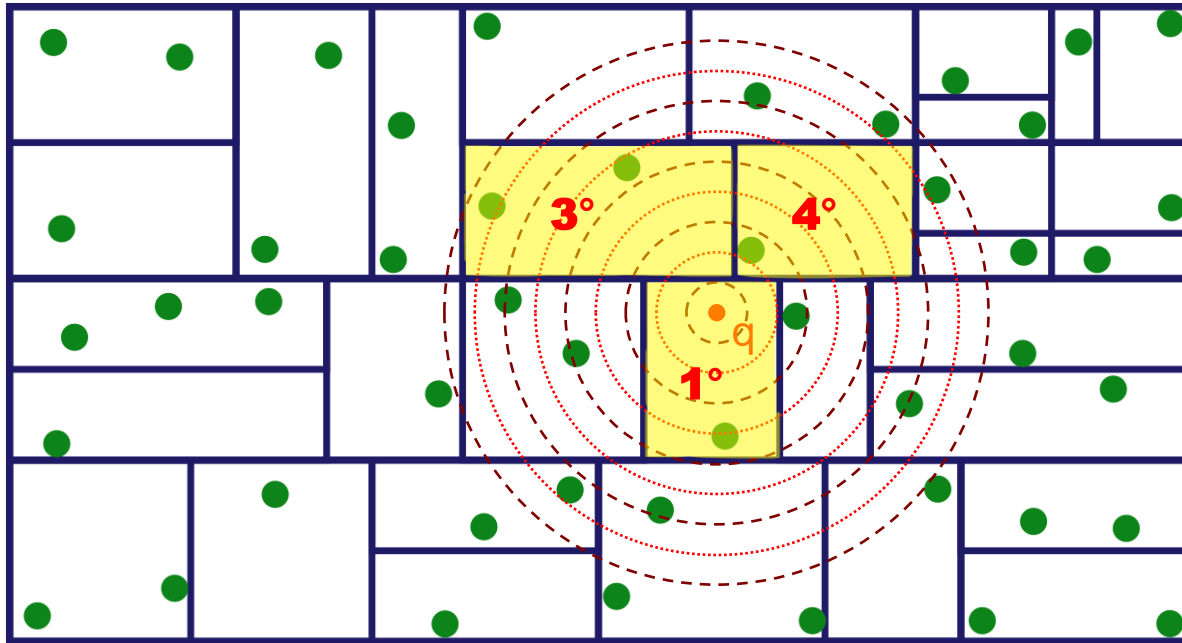


Marcar las regiones hoja que son visitadas para resolver la **búsqueda aproximada del 1-nn usando $c=4$** , el orden en que se visitan esas regiones, los elementos que se les calcula la distancia y señale si el NN encontrado es el correcto o no.

Respuesta Ejercicio 7. Búsqueda aproximada con $c=4$ en randomized k-d trees

- Se visitan las 4 regiones marcadas en amarillo en el orden dado (notar que el 1° y 2° podrían estar intercambiados ya que están al mismo MINDIST (0))
- Se calcula la distancia euclidiana de q a 5 elementos:
 - Un elemento en región 1° + dos elementos en región 2° + dos elementos en región 3° (el elemento de la región 4° ya fue contado en la región 2°)
- El NN encontrado es el correcto.

k-d tree 1



k-d tree 2

