TAREA 2

GEOMETRÍA COMPUTACIONAL 2024.

Las respuestas a esta tarea deberán entregarse antes del primer parcial. Mi sugerencia es que te tome únicamente una semana resolverla. Discutiremos tus avances durante las horas de oficina o después de clase.

(1) Resuelve lo siguiente.

  (a) Sea $S = \{s_1, s_2, \ldots, s_6\}$ el conjunto de segmentos que se muestra en la Figura 1. Denotamos cada segmento s_i como $\overline{a_i, b_i}$ donde a_i es su punto inicial y b_i es su punto final. Al punto de intersección entre el segmento s_i y el segmento s_j lo denotamos como $x_{i,j}$. Demuestra la ejecución del algoritmo Bentley-Ottmann sobre S. Para esto, genera una tabla con las siguientes columnas:

Evento	Pila de Eventos Q	Estatus $\mathcal{L}$	Test de intersección

  En el primer renglón escribe el estado de Q y $\mathcal{L}$ al ser inicializadas por el algoritmo.

  (b) Modifica el algoritmo Bentley-Ottmann para reducir la complejidad espacial del algoritmo a $O(n)$ sin aumentar su complejidad temporal. Usa la siguiente técnica: Borra un punto de intersección de Q cuando los segmentos que lo generan dejan de ser adyacentes (porque otro segmento se encuentra entre ellos). Nota que el mismo punto de intersección es calculado y reinsertado en Q más adelante (si los segmentos vuelven a ser consecutivos). Por ejemplo, considera el punto de intersección $x_{4,5}$ en la Figura 1, en esta nueva versión este punto es eliminado de Q cuando s_6 aparece en el barrido. Al procesar b_6 (el final de s_6), los segmentos s_4 y s_5 vuelven a ser adyacentes y por lo tanto $x_{4,5}$ se vuelve a insertar en Q. Muestra tu trabajo reconstruyendo la tabla del ejercicio anterior con la versión modificada del algoritmo.

(2) Sea S un conjunto de segmentos tales que $S = H \cup V$, donde H es un conjunto de segmentos horizontales y V es un conjunto de segmentos verticales. Diseña un algoritmo que reporte el número de intersecciones. Supón que el conjunto está en posición general. Tu algoritmo debe funcionar en $O(n \lg n)$. Se muestra un ejemplo en la Figura 2.

(3) Sea $\mathcal{S}$ un conjunto de segmentos disjuntos (no se intersectan) y sea $S \in \mathcal{S}$, sea p un punto no en S. Decimos que p *ve totalmente* a S si para todo punto $r \in S$ el segmento $\overline{rp}$ no intersecta a ningún segmento del conjunto $\mathcal{S} \setminus S$. Decimos que p

---

Maestría en Ciencias de la Computación. Cinvestav.

ve parcialmente a S si existe un punto $r \in S$ tal que el segmento $\overline{rp}$ no intersecta a ningún segmento del conjunto $\mathcal{S} \setminus S$. Decimos que *p no ve* a S si no existe ningún punto $r \in S$ tal que el segmento $\overline{rp}$ no intersecta a ningún segmento del conjunto $\mathcal{S} \setminus S$. Por ejemplo, en la Figura 3 el segmento s_1 es totalmente visible desde p, el segmento s_2 es parcialmente visible desde p y el segmento s_3 no es visible desde p.

Diseña un algoritmo que reciba como entrada un conjunto de segmentos $\mathcal{S}$ y un punto p y que genere una etiquetación de los segmentos como parcialmente, totalmente o invisible, dependiendo si p puede o no ver al segmento. Puedes suponer que $p \cap S = \emptyset, \forall S \in \mathcal{S}$. HINT: utiliza la idea del barrido pero esta vez haz un barrido angular.

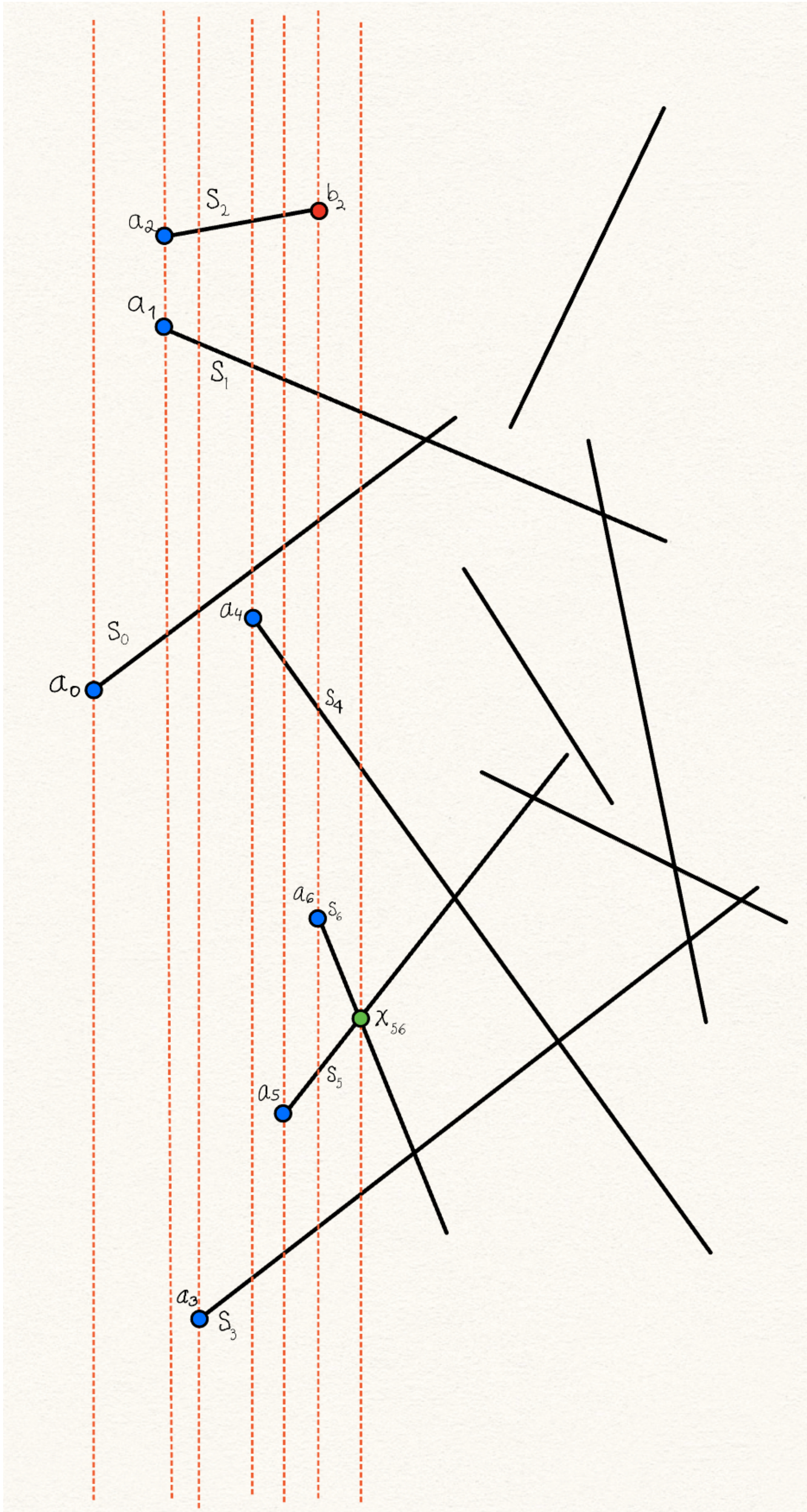

FIGURE 1. Colección de segmentos para el Problema 1.

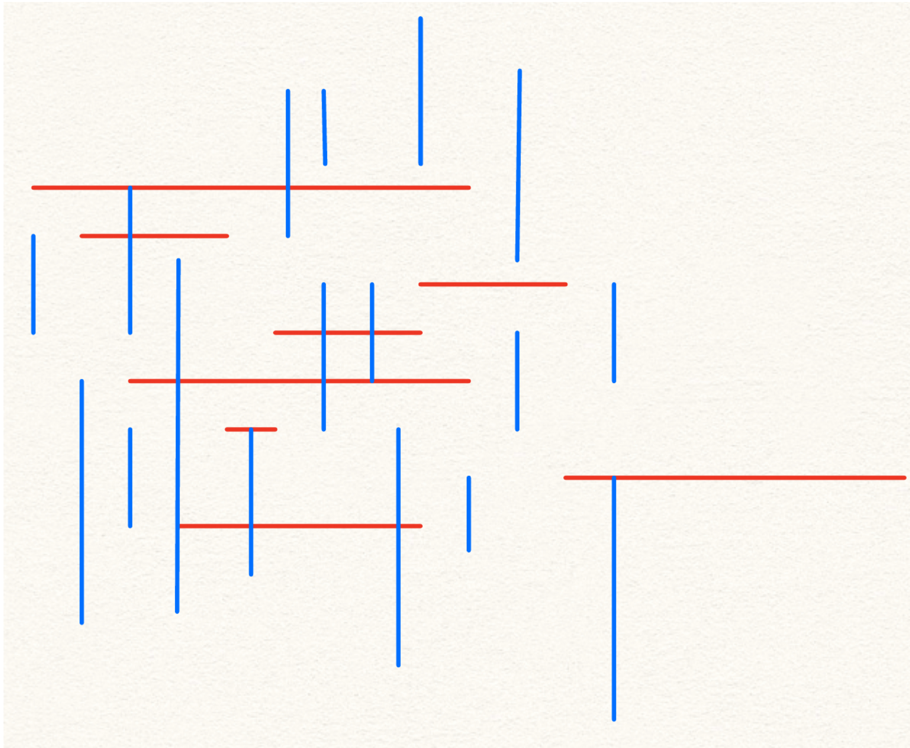

FIGURE 2. Colección de segmentos para el Problema 2.

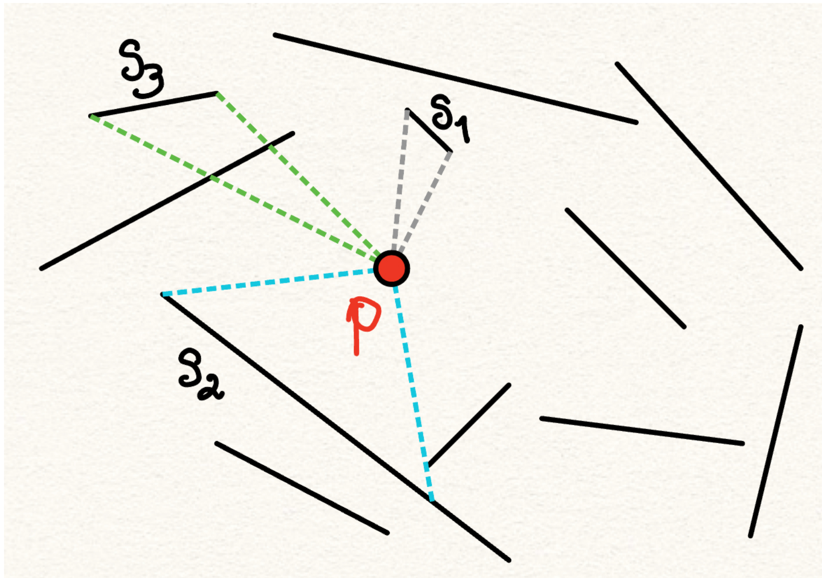

FIGURE 3. Colección de segmentos para el Problema 3.