



**UNIVERSIDAD
DE GRANADA**

decsai.ugr.es

Fundamentos de Programación

Francisco José Cortijo Bon - cb@decsai.ugr.es



DECSAI

**Departamento de Ciencias de la
Computación e Inteligencia Artificial**



**Departamento de Ciencias de la
Computación e Inteligencia Artificial**

Universidad de Granada

**Fundamentos
de Programación**

Francisco José Cortijo Bon

Fundamentos de Programación

Guión de videos

**Relación de
Problemas VI**

Clase Recta

© Francisco José Cortijo Bon
cb@decsai.ugr.es

Última modificación: 18 de Noviembre de 2021

Ejercicios de la relación de problemas 5. Clase *Recta*

(Clase recta) `VI_Recta.cpp`

En este ejercicio se plantean varias modificaciones. Debe entregar un fichero cpp por cada uno de los apartados.

Se desea implementar una `clase Recta` para representar una recta en el plano. Una recta viene determinada por tres coeficientes A, B, C, de forma que todos los puntos (x,y) que pertenecen a la recta verifican lo siguiente (ecuación general de la recta):

$$Ax + By + C = 0$$

Deberá tener en cuenta que A y B no pueden ser simultáneamente 0.

Ejercicios de la relación de problemas 5. Clase *Recta*

(Clase recta) `VI_Recta.cpp`

En este ejercicio se plantean varias modificaciones. Debe entregar un fichero cpp por cada uno de los apartados.

Se desea implementar una clase `Recta` para representar una recta en el plano. Una recta viene determinada por tres coeficientes `A`, `B`, `C`, de forma que todos los puntos (x,y) que pertenecen a la recta verifican lo siguiente (ecuación general de la recta):

$$Ax + By + C = 0$$

Deberá tener en cuenta que `A` y `B` no pueden ser simultáneamente 0.

Ejercicios de la relación de problemas 5. Clase *Recta*

(Clase recta) `VI_Recta.cpp`

En este ejercicio se plantean varias modificaciones. Debe entregar un fichero cpp por cada uno de los apartados.

Se desea implementar una clase `Recta` para representar una recta en el plano. Una recta viene determinada por tres coeficientes `A`, `B`, `C`, de forma que todos los puntos (x,y) que pertenecen a la recta verifican lo siguiente (ecuación general de la recta):

$$Ax + By + C = 0$$

Deberá tener en cuenta que `A` y `B` no pueden ser simultáneamente 0.



Ejercicios de la relación de problemas 5. Clase *Recta*

a) *Definición de la clase y creación de objetos*

Defina la clase *Recta*. En este apartado utilice únicamente datos miembro públicos. Cree un programa principal que haga lo siguiente:

- Defina dos objetos de la clase *Recta*.
- Lea seis reales desde teclado.
- Le asigne los tres primeros a los coeficientes de una recta y los otros tres a la segunda recta.
- Calcule e imprima la pendiente de cada recta aplicando la fórmula:

$$\text{pendiente} = - A / B$$

Ejercicios de la relación de problemas 5. Clase *Recta*

b) *Métodos públicos*

En vez de calcular la pendiente en el programa principal, vamos a ponerlo como un método de la clase y así lo reutilizaremos todas las veces que necesitemos. Añada un método para el cálculo de la pendiente y modifical el `main` para tener en cuenta este cambio. ¿Añadimos pendiente como dato miembro de la recta? La respuesta es que no ¿Por qué? Añadir también los siguiente métodos:

- Método `ToString` para obtener una representación textual (en un `string`) de la ecuación general de la recta.
- Obtener la ordenada (y) dado un valor de abscisa x , aplicando la fórmula:

$$(-C - xA) / B$$
- Obtener la abscisa (x) dado un valor de ordenada y , aplicando la fórmula:

$$(-C - yB) / A$$

En la función `main`, después de reescribir el código de la primera versión usando los métodos propuestos ahora, leed un valor de abscisa e imprimir la ordenada según la recta y leed un valor de ordenada e imprimid la abscisa que le corresponde. Hacedlo sólo con la primera recta.

Ejercicios de la relación de problemas 5. Clase *Recta*

c) *Datos miembro privados*

Cambie ahora los datos miembro públicos y póngalos privados. Tendrá que añadir métodos para asignar y ver los valores de los datos miembro. Añada métodos para asignar un valor a cada uno de los tres datos miembro. Modifique el `main` para tener en cuenta estos cambios.



A partir de ahora, todos los ejercicios deben resolverse utilizando únicamente datos miembro privados.

Ejercicios de la relación de problemas 5. Clase *Recta*

d) *Política de acceso a los datos miembros*

En vez de usar un método para asignar un valor a cada dato miembro, defina un único método SetCoeficientes para asignar los tres a la misma vez. Observad que los métodos permiten definir la política de acceso a los datos miembro.

Si tengo previsto cambiar por separado los coeficientes de la recta, usaré métodos de asignación individuales. En caso contrario, usaré un único método que modifique a la misma vez todos los datos miembro. Incluso pueden dejarse en la clase ambos tipos de métodos para que así el cliente de la clase pueda usar los que estime oportunos en cada momento. Por ahora, mantenga únicamente el método de asignación *en bloque* SetCoeficientes.

Ejercicios de la relación de problemas 5. Clase *Recta*

e) Constructor

Modifique el programa principal del último apartado e imprima los valores de los datos miembros de una recta, **antes** de asignarles los coeficientes. Mostrará, obviamente, un valor indeterminado. Para evitar este problema, añada un constructor a la recta para que el objeto esté en un estado válido en el mismo momento de su definición. El constructor deberá tener como parámetros, obligatoriamente, los tres coeficientes de la recta. Tendrá que modificar convenientemente el `main` para tener en cuenta este cambio.

Ejercicios de la relación de problemas 5. Clase *Recta*

f) *Política de acceso a los datos miembro*

Suprima ahora el método SetCoeficientes. De esta forma, una vez creado el objeto (pasándole los datos apropiados en el constructor) ya no podremos modificar los datos miembro. Esto es útil en aquellas situaciones en las que no queremos permitir que el estado del objeto cambie, una vez se ha creado.

Ejercicios de la relación de problemas 5. Clase *Recta*

g) *Métodos privados*

Añada el método *privado* *Simplifica* para simplificar una recta. El método modificará los coeficientes de la recta dividiéndolos por el máximo común divisor de los tres coeficientes. La recta obtenida será una recta equivalente a la original. Calcule el máximo común divisor con la función `int MCD (int a, int b)` que calcula el máximo común divisor de a y b. Esta función ¿será pública o privada? ¿Por qué? ¿Qué método(s) llamará(n) a *Simplifica*? ¿Tiene sentido que sea público?

Ejercicios de la relación de problemas 5. Clase Recta

Título	VIDEOS
Clase Recta. Apartado a	Clase Recta (Video 1/4)
Clase Recta. Apartados b y c	Clase Recta (Video 2/4)
Clase Recta. Apartados d y e –1ª parte-	Clase Recta (Video 3/4)
Clase Recta. Apartados e –2ª parte- , f y g	Clase Recta (Video 4/4)