

Universidad Nacional del Altiplano
Facultad de Ingeniería Estadística e Informática

Docente:

Ing. Torres Cruz Fred

Alumno:

Ticona Miramira Roberto Angel

Trabajo 3 - Restricciones y Sistemas de Ecuaciones

» DESCRIPCIÓN

Las restricciones son condiciones que limitan el conjunto de valores posibles que pueden tomar las variables en un problema. Estas son esenciales en problemas de optimización y en sistemas de ecuaciones, ya que definen la región factible, es decir, el espacio de soluciones posibles. En ese sentido se proponen 5 ejercicios que deben ser desarrollados utilizando las restricciones y el código del anterior trabajo para realizar una representación gráfica.

Aplicación en ejercicios

1.- Un desarrollador tiene 15 horas semanales para dedicar al desarrollo de software de front-end (x) y back-end (y). Además:

- Debe dedicar al menos 5 horas al desarrollo de front-end para cumplir con los entregables del cliente.
- El tiempo total no puede exceder 15 horas por restricciones de tiempo del sprint.

Formule las restricciones, represéntelas gráficamente e identifique las combinaciones posibles de tiempo a invertir en cada actividad.

Variables

- x = horas dedicadas al front-end
- y = horas dedicadas al back-end

Restricciones

1. Debe dedicar al menos 5 horas al mes.
 - $x \geq 5$
2. El tiempo total no puede exceder a 15 horas.
 - $x + y \leq 15$

Gráfico generado con python



Interpretación del gráfico

En la representación gráfica se observa la región factible determinada por las restricciones planteadas:

- $x \geq 5$: el desarrollador debe dedicar al menos 5 horas al front-end. Esta condición aparece como una línea vertical en $x = 5$ y la región válida se encuentra a la derecha de ella.
- $y \geq 0$: el tiempo dedicado al back-end no puede ser negativo, por lo que la región se limita a la parte superior del eje x .
- $x + y \leq 15$: la suma de horas asignadas a front-end y back-end no puede superar las 15 horas semanales. Gráficamente, corresponde a la semirrecta bajo la línea $x + y = 15$.

La intersección de estas tres restricciones genera una región triangular factible delimitada por los puntos (5, 0), (15, 0) y (5, 10). Esto significa que cualquier combinación de horas ubicada dentro o sobre este triángulo cumple con las condiciones del problema. Por ejemplo, el desarrollador podría dedicar:

- 5 horas a front-end y 10 horas a back-end,
- 10 horas a front-end y 5 horas a back-end,
- o bien 15 horas únicamente a front-end.

En conclusión, la región sombreada representa todas las combinaciones posibles de tiempo de trabajo entre front-end y back-end que respetan tanto el mínimo requerido en front-end como la restricción máxima de 15 horas semanales.

Modificación del código original

Para poder generar la función el código original se modificó y añadió algunas partes.

```

1 # Función para preparar expresiones lineales
2 def preparar_expresion(expr: str) -> str:
3     expr = expr.replace(" ", "")           # quitar espacios
4     expr = expr.replace("^", "**")         # potencia
5     expr = expr.replace("-x", "-1*x")      # caso -x
6     expr = expr.replace("+x", "+1*x")      # caso +x
7     if expr.startswith("x"):              # si empieza con x
8         expr = "1*" + expr
9     expr = expr.replace("x", "*x")         # poner multiplicación
10    expr = expr.replace("**x", "*x")       # corregir si se duplicó
11    return expr
12
13 # Restricciones:
14 # 1) x = 5 (vertical)
15 # 2) x + y = 15 -> y = -x + 15
16 func2 = preparar_expresion("-x+15")
17
18 # Rango de la gráfica
19 xmin, xmax = -5, 20
20 ymin, ymax = -5, 20
21
22 # Recorremos el plano
23 for y in range(ymax, ymin - 1, -1):
24     linea = ""
25     for x in range(xmin, xmax + 1):
26         # Recta 1: x = 5
27         cond1 = (x == 5)
28
29         # Recta 2: y = -x + 15
30         try:
31             y2 = eval(func2)
32         except:
33             y2 = None
34         cond2 = (y2 is not None and abs(y - y2) < 0.5)
35
36         # Región factible: x>=5, y>=0, x+y<=15
37         region = (x >= 5 and y >= 0 and x + y <= 15)
38
39         # Qué dibujar
40         if cond1 and cond2:
41             linea += "#"
42         elif cond1:
43             linea += "*"
44         elif cond2:
45             linea += "o"

```

```

46         elif x == 0 and y == 0:
47             linea += "+"
48         elif x == 0:
49             linea += "|"
50         elif y == 0:
51             linea += "-"
52         elif region:
53             linea += "."
54         else:
55             linea += " "
56     print(linea)
57
58 # Leyenda
59 print("\nLeyenda del gráfico:")
60 print(" * = x = 5")
61 print(" o = x + y = 15")
62 print(" # = Intersección")
63 print(" . = Región factible")
64 print(" | = Eje Y")
65 print(" - = Eje X")
66 print(" + = Origen (0,0)")

```

2.- Un ingeniero de datos administra dos tipos de servidores en la nube: Servidores A y Servidores B. El costo por hora de Servidor A es $S/3$ y de Servidor B es $S/5$. El presupuesto máximo semanal asignado para mantener los servidores es de $S/20$. Determine cuántas horas puede mantener activos cada tipo de servidor, formule el sistema de ecuaciones y represéntelo gráficamente.

Variables

- x = horas de uso de Servidor A
- y = horas de uso de Servidor B

Restricción

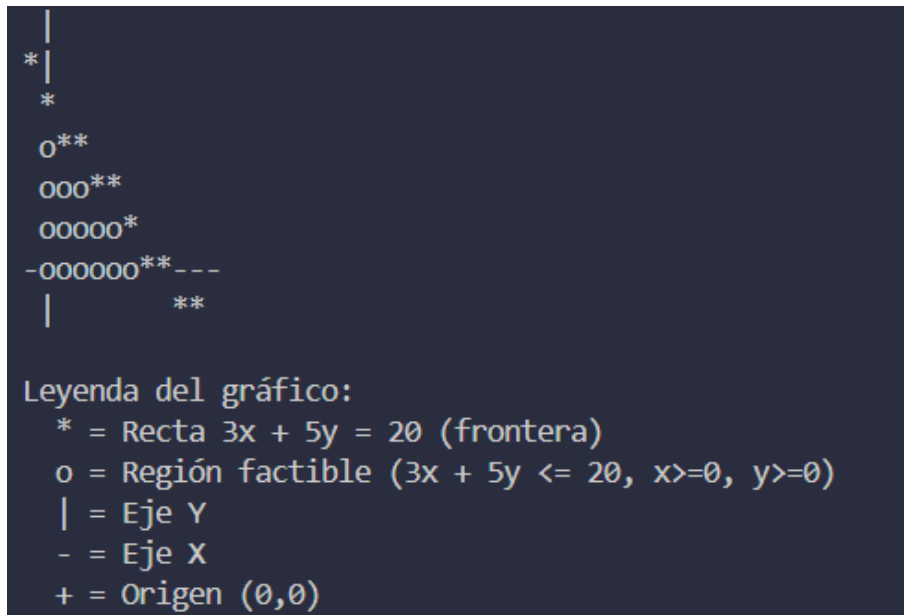
1. Se sabe que la hora de Servidor A cuesta $S/3$, cada hora de Servidor B cuesta $S/5$ y el presupuesto máximo es $S/20$.

- $3x + 5y \leq 20$

Sistema de ecuaciones lineales

$$\begin{cases} 3x + 5y \leq 20 \\ x \geq 0 \\ y \geq 0 \end{cases}$$

Gráfico generado con Python



Interpretación del gráfico

La recta límite está dada por $3x + 5y = 20$. Al interceptar con los ejes coordenados se obtiene que, si $x = 0$, entonces $y = 4$, y si $y = 0$, entonces $x = \frac{20}{3} \approx 6.67$.

La región factible corresponde al triángulo delimitado por los puntos $(0,0)$, $(6.67,0)$ y $(0,4)$. Cualquier combinación de horas (x,y) dentro de esta región satisface las condiciones de costo máximo.

En conclusión, el ingeniero de datos puede asignar:

- hasta 6 horas con 40 minutos únicamente a servidores A,
- hasta 4 horas únicamente a servidores B,
- o bien cualquier combinación intermedia que cumpla con el presupuesto, como por ejemplo $(x = 2, y = 2)$ o $(x = 3, y = 1)$.

Modificación del código original

Para poder generar la función el código original se modificó y añadió algunas partes.

```

1 # Restricción principal: 3x + 5y <= 20 → y <= (20 - 3x)/5
2 func = "(20 - 3*x)/5"
3
4 # Definimos el rango del gráfico (valores de X y Y)
5 xmin, xmax = -1, 10      # un poco más para ver bien
6 ymin, ymax = -1, 6
7
8 # Recorremos los valores de Y de arriba hacia abajo
9 for y in range(ymax, ymin - 1, -1):
10     linea = ""
11
12     for x in range(xmin, xmax + 1):
13         try:
14             y_line = eval(func)
15         except:
16             y_line = None
17
18     # Condición para la recta
19     cond_line = (y_line is not None and abs(y - y_line) < 0.5)

```

```

20
21     # Condición para la región factible: debajo de la recta y en el primer
        ↳ cuadrante
22     cond_region = (y_line is not None and y <= y_line and x >= 0 and y >= 0)
23
24     if cond_line:
25         linea += "*"
26     elif cond_region:
27         linea += "o"
28     elif x == 0 and y == 0:
29         linea += "+"
30     elif x == 0:
31         linea += "|"
32     elif y == 0:
33         linea += "-"
34     else:
35         linea += " "
36     print(linea)
37
38 # Leyenda
39 print("\nLeyenda del gráfico:")
40 print(" * = Recta 3x + 5y = 20 (frontera)")
41 print(" o = Región factible (3x + 5y <= 20, x>=0, y>=0)")
42 print(" | = Eje Y")
43 print(" - = Eje X")
44 print(" + = Origen (0,0)")

```

3.- Un administrador de proyectos tecnológicos organiza su tiempo entre reuniones con stakeholders (x) y trabajo en la documentación técnica (y). Las reuniones requieren al menos 4 horas semanales y la documentación al menos 6 horas. Si dispone de 12 horas para ambas actividades, determine la región factible y analice las combinaciones posibles de tiempo.

Variables

- x = horas dedicadas a reuniones con stakeholders
- y = horas dedicadas a la documentación técnica

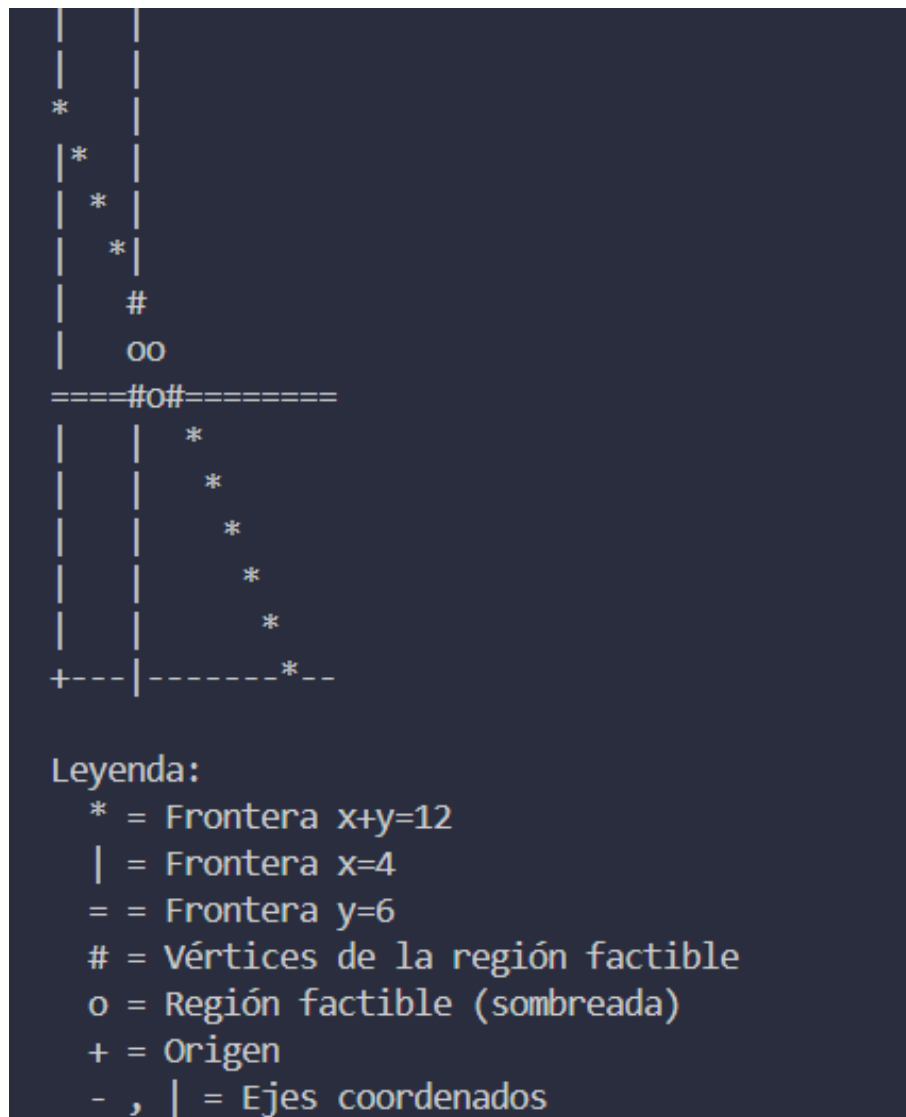
Condiciones

1. Reuniones de al menos 4 horas
 - $x \geq 4$
2. Documentación de al menos 6 horas
 - $y \geq 6$
3. Tiempo total disponible máximo de 12 horas
 - $x + y \leq 12$

Sistema de ecuaciones lineales

$$\begin{cases} x \geq 4 \\ y \geq 6 \\ x + y \leq 12 \end{cases}$$

Gráfico generado con Python



Interpretación del gráfico

La recta límite es $x + y = 12$, que corta los ejes en los puntos $(12, 0)$ y $(0, 12)$. Aplicando las condiciones $x \geq 4$ y $y \geq 6$, la región factible se reduce al triángulo formado por los puntos:

$$(4, 6), \quad (4, 8), \quad (6, 6).$$

Esto significa que el administrador debe dedicar al menos 4 horas a reuniones y 6 horas a documentación, pero no más de 12 horas en total.

En consecuencia, puede combinar su tiempo de distintas formas dentro de ese triángulo, por ejemplo:

- $(x = 4, y = 7)$: 4 horas de reuniones y 7 de documentación.
- $(x = 5, y = 6)$: 5 horas de reuniones y 6 de documentación.
- $(x = 6, y = 6)$: combinación en el límite máximo permitido.

Modificación del código original

Para poder generar la función el código original se modificó y añadió algunas partes.

```

1 # Graficar restricciones del problema 3 en ASCII con sombreado de la región factible
2
3 xmin, xmax = 0, 14
4 ymin, ymax = 0, 14
5
6 for y in range(ymax, ymin - 1, -1):
7     linea = ""
8     for x in range(xmin, xmax + 1):
9
10         # Restricciones
11         cond_recta = (abs(x + y - 12) < 0.5)    # frontera x+y=12
12         cond_x = (x == 4)                      # frontera x=4
13         cond_y = (y == 6)                      # frontera y=6
14
15         # Región factible: x>=4, y>=6, x+y<=12
16         cond_region = (x >= 4 and y >= 6 and (x + y <= 12))
17
18         # Puntos de intersección (vértices)
19         vertices = [(4,6), (4,8), (6,6)]
20         cond_vertice = (x,y) in vertices
21
22         # Dibujar con prioridad
23         if cond_vertice:
24             linea += "#"
25         elif cond_region:
26             linea += "o"
27         elif cond_recta:
28             linea += "*"
29         elif cond_x:
30             linea += "|"
31         elif cond_y:
32             linea += "="
33         elif x == 0 and y == 0:
34             linea += "+"
35         elif x == 0:
36             linea += "|"
37         elif y == 0:
38             linea += "-"
39         else:
40             linea += " "
41     print(linea)
42
43 print("\nLeyenda:")
44 print(" * = Frontera x+y=12")
45 print(" | = Frontera x=4")
46 print(" = = Frontera y=6")
47 print(" # = Vértices de la región factible")
48 print(" o = Región factible (sombreada)")
49 print(" + = Origen")
50 print(" - , | = Ejes coordenados")

```

4.- Una empresa de desarrollo de videojuegos produce dos tipos de assets: Modelos 3D (P1) y Texturas (P2). Cada modelo 3D requiere 2 horas de trabajo y cada textura requiere 3 horas. El equipo de arte tiene un total de 18 horas disponibles semanalmente. Formule las restricciones, represéntelas gráficamente y determine cuántos assets de cada tipo pueden producirse en función del tiempo disponible.

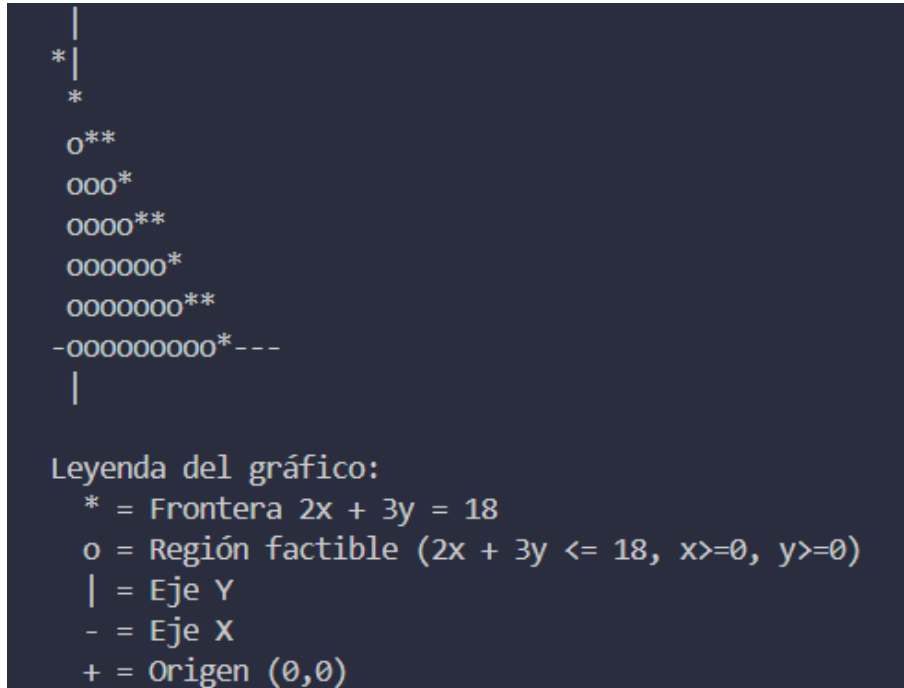
Variables

- x = número de modelos 3D (P1)
- y = número de texturas (P2)

Restricciones

- Restricción principal: $2x + 3y \leq 18$
- Restricciones adicionales: $x \geq 0$, $y \geq 0$

Gráfico generado con Python



Interpretación del gráfico

La frontera de la región factible está dada por la recta $2x + 3y = 18$, que corta al eje x en el punto (9,0) y al eje y en (0,6). La región factible corresponde al triángulo con vértices (0,0), (9,0) y (0,6).

El equipo puede producir hasta 9 modelos 3D si dedica todo el tiempo a ellos, o hasta 6 texturas si dedica todo a ese tipo de asset. También puede realizar combinaciones intermedias, siempre que se cumpla $2x + 3y \leq 18$. Por ejemplo, 3 modelos y 4 texturas consumen exactamente las 18 horas disponibles, mientras que 5 modelos y 2 texturas consumen 16 horas, quedando dentro de la región factible.

Modificación del código original

Para poder generar la función el código original se modificó y añadió algunas partes.

```

1 # Problema 4 - Modelos 3D y Texturas
2
3 # Definimos el rango del gráfico
4 xmin, xmax = -1, 12
5 ymin, ymax = -1, 8
6
7 # Recorremos los valores de Y de arriba hacia abajo
8 for y in range(ymax, ymin - 1, -1):
9     linea = ""
10
11     for x in range(xmin, xmax + 1):
12
13         # Ecuación de la frontera: 2x + 3y = 18
14         y_line = (18 - 2*x) / 3 if (18 - 2*x) >= 0 else None

```

```
15
16     # Condición para estar en la recta
17     cond_line = (y_line is not None and abs(y - y_line) < 0.5)
18
19     # Condición para estar en la región factible
20     cond_region = (x >= 0 and y >= 0 and (2*x + 3*y <= 18))
21
22     if cond_line:
23         linea += "*"
24     elif cond_region:
25         linea += "o"
26     elif x == 0 and y == 0:
27         linea += "+"
28     elif x == 0:
29         linea += "|"
30     elif y == 0:
31         linea += "-"
32     else:
33         linea += " "
34     print(linea)
35
36 # Leyenda
37 print("\nLeyenda del gráfico:")
38 print(" * = Frontera 2x + 3y = 18")
39 print(" o = Región factible (2x + 3y <= 18, x>=0, y>=0)")
40 print(" | = Eje Y")
41 print(" - = Eje X")
42 print(" + = Origen (0,0)")
```

5.- Una startup de hardware dispone de un máximo de 50 unidades de componentes electrónicos. Para ensamblar un dispositivo tipo A se necesitan 5 unidades y para un dispositivo tipo B se necesitan 10 unidades. Determine cuántos dispositivos de cada tipo puede ensamblar sin exceder las 50 unidades de componentes. Formule el problema, resuélvalo gráficamente y explique las posibles combinaciones de producción.

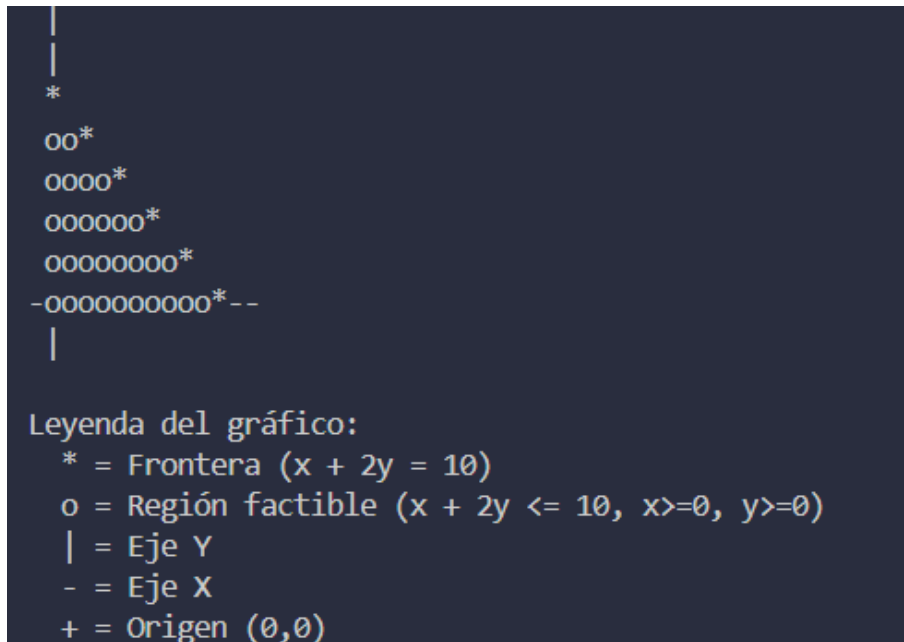
Variables

- x = número de dispositivos tipo A
- y = número de dispositivos tipo B

Restricciones

- Restricción principal: $5x + 10y \leq 50$
- Restricciones adicionales: $x \geq 0, y \geq 0$

Gráfico generado con Python



Interpretación

La frontera de la región factible está dada por la recta $5x + 10y = 50$, equivalente a $x + 2y = 10$. Esta recta corta al eje x en el punto $(10, 0)$ y al eje y en $(0, 5)$. La región factible corresponde al triángulo con vértices $(0, 0)$, $(10, 0)$ y $(0, 5)$.

La empresa puede ensamblar hasta 10 dispositivos tipo A si dedica todos los componentes a ellos, o hasta 5 dispositivos tipo B si dedica todos los componentes a ese tipo. También puede realizar combinaciones intermedias, siempre que se cumpla $5x + 10y \leq 50$. Por ejemplo, 6 dispositivos tipo A y 2 tipo B utilizan exactamente las 50 unidades, al igual que 4 dispositivos tipo A y 3 tipo B.

Modificación del código original

Para poder generar la función el código original se modificó y añadió algunas partes.

```

1 # Problema 5 - Dispositivos A y B
2
3 # Definimos el rango del gráfico
4 xmin, xmax = -1, 12
5 ymin, ymax = -1, 7
6
7 # Recorremos los valores de Y de arriba hacia abajo
8 for y in range(ymax, ymin - 1, -1):
9     linea = ""
10
11     for x in range(xmin, xmax + 1):
12
13         # Recta de frontera:  $x + 2y = 10 \rightarrow y = (10 - x)/2$ 
14         y_line = (10 - x) / 2 if (10 - x) >= 0 else None
15
16         # Condición para estar en la recta
17         cond_line = (y_line is not None and abs(y - y_line) < 0.5)
18
19         # Condición para estar en la región factible
20         cond_region = (x >= 0 and y >= 0 and (x + 2*y <= 10))
21
22         if cond_line:
23             linea += "*"

```

```
24         elif cond_region:
25             linea += "o"
26         elif x == 0 and y == 0:
27             linea += "+"
28         elif x == 0:
29             linea += "|"
30         elif y == 0:
31             linea += "-"
32         else:
33             linea += " "
34     print(linea)
35
36 # Leyenda
37 print("\nLeyenda del gráfico:")
38 print(" * = Frontera ( $x + 2y = 10$ )")
39 print(" o = Región factible ( $x + 2y \leq 10$ ,  $x \geq 0$ ,  $y \geq 0$ )")
40 print(" | = Eje Y")
41 print(" - = Eje X")
42 print(" + = Origen (0,0)")
```