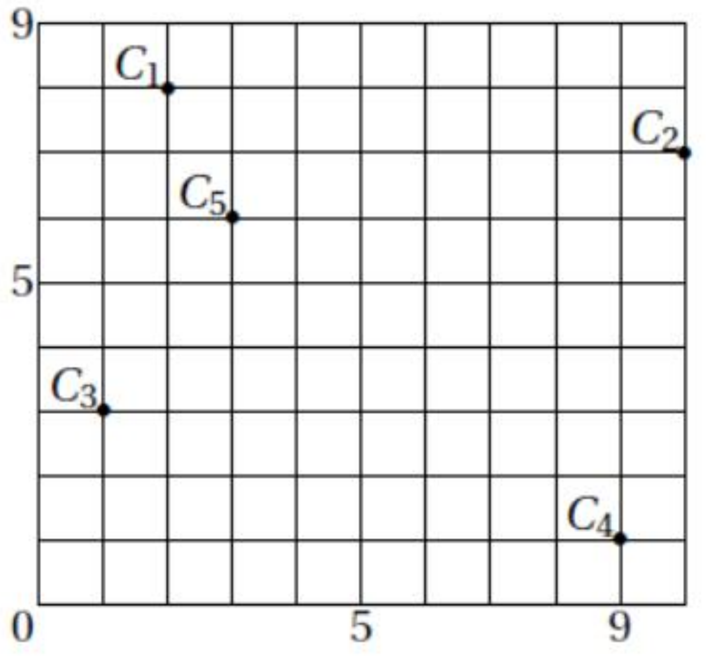
8. O problema de localização consiste em seleccionar os locais onde devem ser estabelecidas instalações de forma a melhor servir um dado conjunto de clientes. Existem 5 clientes, designados por $C_1, \ldots, C_5$, nos locais indicados na figura, com coordenadas $(2,8)^\top, (10,7)^\top, (1,3)^\top$, $(9,1)^\top$ e $(3,6)^\top$, respectivamente.



Considere que a distância entre os pontos é medida de uma forma rectilínea, ao longo de linhas verticais e horizontais, naquela que é, por vezes, designada por distância de Manhattan. Dados dois pontos $(x_1, y_1)^\top$ e $(x_2, y_2)^\top$, a distância entre eles é dada por $|x_1 - x_2| + |y_1 - y_2|$.

Existem muitas variantes do problema de localização, com diferentes tipos de função objectivo e de restrições. Formule os seguintes problemas usando programação linear. Explique detalhadamente o significado das variáveis de decisão e das restrições do modelo.

a) Decidir a localização de um depósito, designado por D, que vai servir os 5 clientes, com o objectivo de minimizar a soma dos custos de transporte, o que ocorre, por exemplo, em sistemas de distribuição de mercadorias. Os custos de transporte são calculados multiplicando a distância entre o depósito e o cliente pelo número de deslocações mensais, dado pela seguinte tabela:

	C1	C2	C3	C4	C5
# desl.	15	8	17	12	4

b) Decidir a localização de uma instalação, designada por B, com o objectivo de minimizar a maior distância entre B e o cliente mais distante, o que ocorre na localização de serviços de emergência, como, por exemplo, de hospitais ou de bombeiros, em que se pretende minimizar o tempo máximo que decorre até se iniciar a prestação do serviço.

a) Decidir a localização de um depósito, designado por D, que vai servir os 5 clientes, com o objectivo de minimizar a soma dos custos de transporte, o que ocorre, por exemplo, em sistemas de distribuição de mercadorias. Os custos de transporte são calculados multiplicando a distância entre o depósito e o cliente pelo número de deslocações mensais, dado pela seguinte tabela:

	C1	C2	C3	C4	C5
# desl.	15	8	17	12	4

Dados
- Coordenadas dos clientes; número de viagens mensais

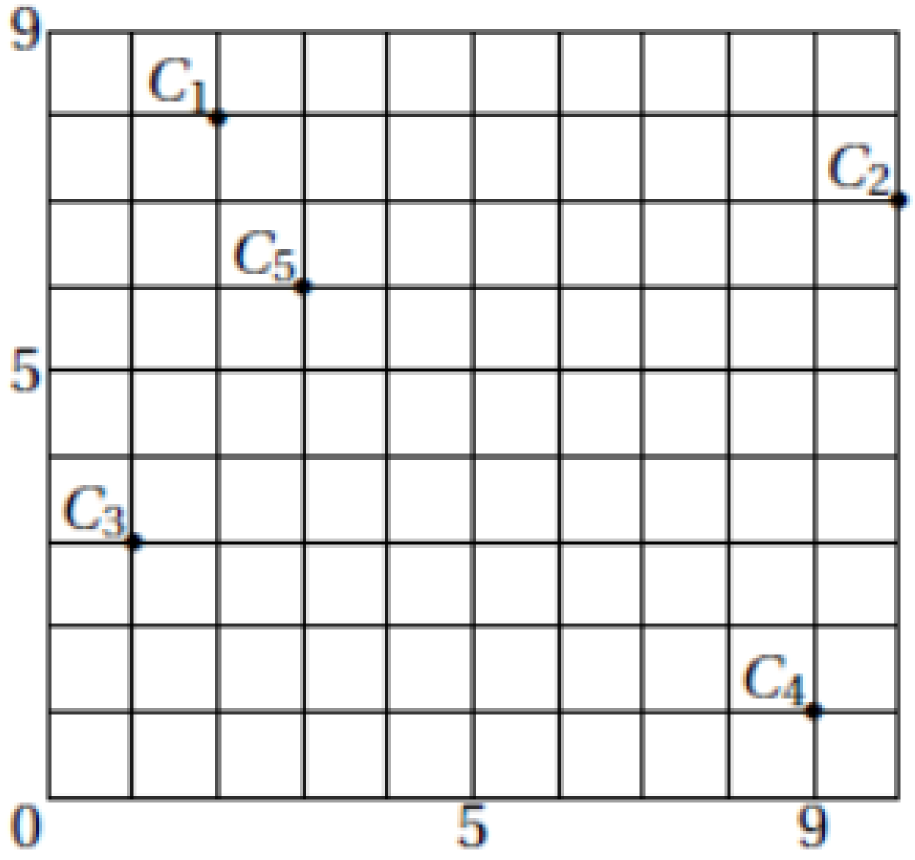
Variáveis de decisão
- x, y: coordenadas do depósito D

Restrições
- colocação num ponto da grelha: 0 <= x <= 10, 0 <= y <= 9 (serão precisas?)
- funções que explicitam a distância de cada cliente ao depósito (estas funções vão usar variáveis de decisão adicionais)

Função objectivo
- Minimizar soma das distâncias aos clientes pesadas com o número de viagens

Variáveis de decisão
- x, y: coordenadas do depósito D

NOTA:
- as variáveis x e y podem ser fraccionárias;
- se se pretender que o local fique na intersecção de 2 traços do quadriculado, as variáveis x e y devem ser declaradas como inteiras.



funções que explicitam a distância de cada cliente ao depósito

- O cliente j tem coordenadas (coord_xj ,coord_yj)

- A distância do cliente j ao depósito ao longo do eixo dos x é
  dada pelo valor absoluto de: x - coord_xj, | x - coord_xj |

- O valor de (x - coord_xj) pode ser positivo ou negativo,
consoante a posição relativa do cliente e do depósito.

Usa-se uma variável sem restrição de sinal para representar (x - coord_xj).

- Uma variável sem restrição de sinal é expressa como a diferença entre duas variáveis não negativas:
- (x - coord_xj) = xjp - xjn

- xjp expressa o valor de (x - coord_xj) quando (x - coord_xj) >= 0
- xjn expressa o valor de -(x - coord_xj) quando (x - coord_xj) < 0

funções que explicitam a distância de cada cliente ao depósito

- A função xjp + xjn expressa o valor absoluto da distância entre o depósito e o cliente j ao longo do eixo x.

- xjp + xjn = |xjp, se a posição relativa for positiva
  |xjn, se a posição relativa for negativa

(quando um é positivo, espera-se que o outro seja = 0, porquê?)

- o mesmo para a coordenada y

- dj = xjp + xjn + yjp + yjn = distância total entre o depósito e o cliente j (eixos x e y)

funções que explicitam a distância de cada cliente ao depósito

Exemplo: cliente 1, com coordenadas (2,8)T

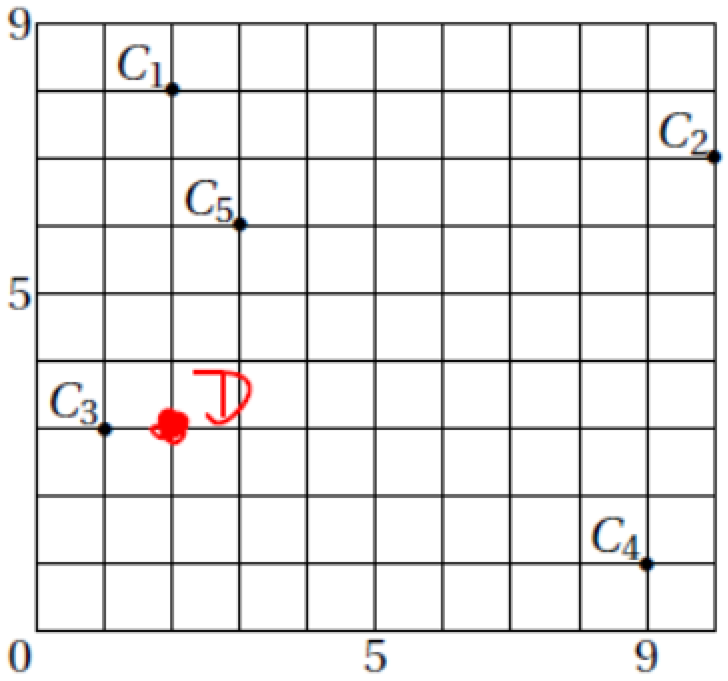
```
// posição relativa ao longo do eixo do x
x - 2 = x1p - x1n;

// posição relativa ao longo do eixo do y
y - 8 = y1p - y1n;

// distância total (eixos x e y) entre o local e o cliente j
x1p + x1n + y1p + y1n = d1;
```

```
min:  15 d1 +  8 d2 + 17 d3 +  12 d4 +  4 d5;

// posição relativa ao longo do eixo do x
x - 2 = x1p - x1n;
x -10 = x2p - x2n;
x - 1 = x3p - x3n;
x - 9 = x4p - x4n;
x - 3 = x5p - x5n;

// posição relativa ao longo do eixo do y
y - 8 = y1p - y1n;
y - 7 = y2p - y2n;
y - 3 = y3p - y3n;
y - 1 = y4p - y4n;
y - 6 = y5p - y5n;

// distância total (eixos x e y) entre o local e o cliente j
x1p + x1n + y1p + y1n = d1;
x2p + x2n + y2p + y2n = d2;
x3p + x3n + y3p + y3n = d3;
x4p + x4n + y4p + y4n = d4;
x5p + x5n + y5p + y5n = d5;
```

Solução óptima e verificação da solução

cliente	distância	#viagens	Dist.total
1	5	15	75
2	12	8	96
3	1	17	17
4	9	12	108
5	4	4	16
total			312

	C1	C2	C3	C4	C5
# desl.	15	8	17	12	4



Variables	result
	312
d2	12
d4	9
x2n	8,00000000000001
x4n	7
d1	5
y1n	5
d5	4,00000000000001
y2n	4
y5n	3,00000000000001
y	3
y4p	2
x	1,99999999999999
x5n	1
d3	0,999999999999993
x3p	0,999999999999992

b) Decidir a localização de uma instalação, designada por B, com o objectivo de minimizar a maior distância entre B e o cliente mais distante, o que ocorre na localização de serviços de emergência, como, por exemplo, de hospitais ou de bombeiros, em que se pretende minimizar o tempo máximo que decorre até se iniciar a prestação do serviço.

Dados
- Coordenadas dos clientes

Variáveis de decisão
- x, y: coordenadas da instalação B

Restrições
- colocação num ponto da grelha: portanto 0 <= x <= 10, 0 <= y <= 9
- funções que explicitam a distância de cada cliente à instalação B (estas funções vão usar variáveis de decisão adicionais)

Função objectivo
- Minimizar a maior das distâncias dos clientes à instalação B

Já temos as funções que explicitam a distância de cada cliente ao ponto B:

- dj = xjp + xjn + yjp + yjn = distância total entre o depósito e o cliente j (eixos x e y)

Como Minimizar a maior das distâncias ?

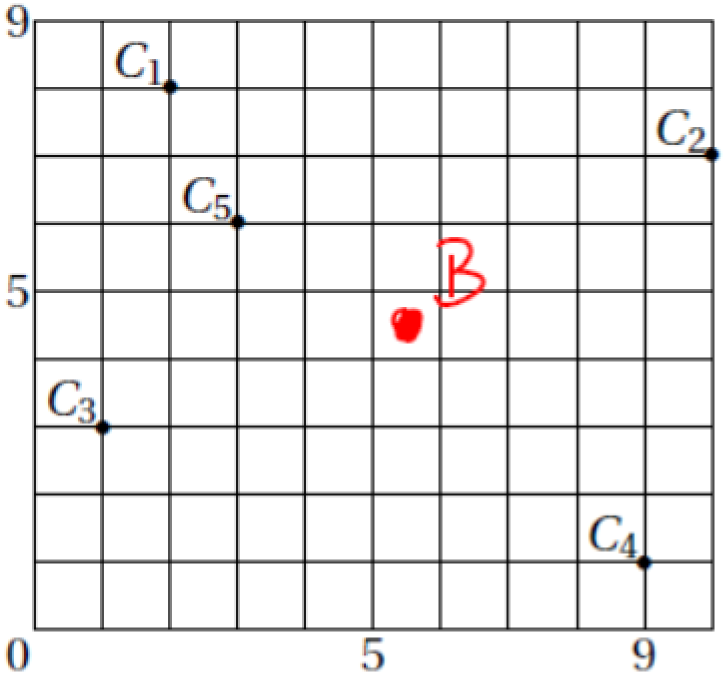
```
// encontrar o maior valor de distância
dmax >= d1;
dmax >= d2;
dmax >= d3;
dmax >= d4;
dmax >= d5;
```

```
min:  dmax;
```

```
min:  dmax;
// posição relativa ao longo do eixo do x
x - 2 = x1p - x1n;
x -10 = x2p - x2n;
x - 1 = x3p - x3n;
x - 9 = x4p - x4n;
x - 3 = x5p - x5n;
// posição relativa ao longo do eixo do y
y - 8 = y1p - y1n;
y - 7 = y2p - y2n;
y - 3 = y3p - y3n;
y - 1 = y4p - y4n;
y - 6 = y5p - y5n;
// distância total (eixos x e y) entre o local e o cliente j
x1p + x1n + y1p + y1n = d1;
x2p + x2n + y2p + y2n = d2;
x3p + x3n + y3p + y3n = d3;
x4p + x4n + y4p + y4n = d4;
x5p + x5n + y5p + y5n = d5;
// encontra o maior valor de distância
dmax >= d1;
dmax >= d2;
dmax >= d3;
dmax >= d4;
dmax >= d5;
```

Solução óptima e verificação da solução

cliente	Distância x	Distância y	di
1	3.5	3.5	7
2	4.5	2.5	7
3	4.5	1.5	6
4	3.5	3.5	7
5	2.5	1.5	4
min max			7



Uma variável sem restrição de sinal é definida por:

x = xp - xn , xp, xn >= 0.

O uso de uma penalidade infinitesimal na função objectivo para a variável x não altera a solução óptima e evita que as 2 variáveis xp e xn sejam ambas positivas.

<u>Exemplo:</u>

Se x5p = 4 e x5n = 1.5 (ambas positivas), a distância d5 = x5p + x5n + y5p + y5n = 4 + 1.5 + 0 + 1.5 = 7.

Seria uma solução óptima alternativa admissível, porque o valor óptimo da função objectivo seria também 7.

Variables	result
	7,000...
d1	7
d4	7
d2	7
dmax	7
d3	6
x	5,5
x3p	4,5
y	4,5
x2n	4,5
d5	4
x1p	3,5
y4p	3,5
y1n	3,5
x4n	3,5
x5p	2,5
y2n	2,5
y3p	1,5
y5n	1,5
y3n	0

```
min:  dmax + 1e-10 d1 + 1e-10 d2 + 1e-10 d3 + 1e-10 d4 + 1e-10 d5;
// posição relativa ao longo do eixo do x
x - 2 = x1p - x1n;
x -10 = x2p - x2n;
x - 1 = x3p - x3n;
x - 9 = x4p - x4n;
x - 3 = x5p - x5n;
// posição relativa ao longo do eixo do y
y - 8 = y1p - y1n;
y - 7 = y2p - y2n;
y - 3 = y3p - y3n;
y - 1 = y4p - y4n;
y - 6 = y5p - y5n;
// distância total (eixos x e y) entre o local e o cliente j
x1p + x1n + y1p + y1n = d1;
x2p + x2n + y2p + y2n = d2;
x3p + x3n + y3p + y3n = d3;
x4p + x4n + y4p + y4n = d4;
x5p + x5n + y5p + y5n = d5;
// encontra o maior valor de distância
dmax >= d1;
dmax >= d2;
dmax >= d3;
dmax >= d4;
dmax >= d5;
```