Homework #4
Carmen Schuler

1.

dept	age	salary	status	count
sales	31-35	46K-50K	senior	30
sales	26-30	26K-30K	junior	40
sales	31-35	31K-35K	junior	40
systems	21-25	46K-50K	junior	20
systems	31-35	66K-70K	senior	5
systems	26-30	46K-50K	junior	3
systems	41-45	66K-70K	senior	3
marketing	36-40	46K-50K	senior	10
marketing	31-35	41K-45K	junior	4
secretary	46-50	36K-40K	senior	4
secretary	26-30	26K-30K	junior	6
				165

Data to be classified:
X = (department = systems, age = 26-30, salary = 46K-50K)

P(Ci): P(junior) = 113/165 = 0.685
P(senior) = 52/165 = 0.315

P(X/Ci) for each class:
P(dept = systems / junior) = 23/113 = 0.204
P(age = 26-30/junior) = 49/113 = 0.434
P(salary = 46K-50K/junior) = 23/113 = 0.204
P(dept = systems / senior) = 8/52 = 0.154
P(age = 26-30/senior) = 0/52 = 0
this cannot be 0 so use laplacian correction:
possible cases for age = 6 (21-25, 26-30, 31-35, 36-40, 41-45, 46-50)
=C(age=26-30) + 1 / C(senior)+(1*6) = (0+1) / (52 + 6) = 1/58 = 0.017
new P(age = 26-30/senior) = 0.017
P(salary = 46K-50K/senior) = 40/52 = 0.769

P(X/Ci): P(X/junior) = 0.204 * 0.434 * 0.204 = 0.018
P(X/senior) = 0.154 * 0.017 * 0.769 = 0.002

P(X/Ci) * P(Ci):
P(X/junior) * P(junior) = 0.018 * 0.685 = 0.0123
P(X/senior) * P(senior) = 0.002 * 0.315 = 0.00063

Therefore, X belongs to class "junior"

2. See Python Notebook

3a. Initial centroid: c: (8, 4)
h: (4, 9)

	Data objects (x1, x2)		Distance to c		Distance to h		Cluster
			8	4	4	9	
a	2	10	12		3		h
b	2	5	7		6		h
c	8	4	0		9		c
d	5	8	7		2		h
e	7	5	2		7		c
f	6	4	2		7		c
g	1	2	9		10		c
h	4	9	9		0		h

c cluster = c, e, f, g
h cluster = a, b, d, h

c cluster centroid: x1 = (8+7+6+1)/4 = 5.5
x2 = (4+5+4+2)/4 = 3.75

h cluster centroid: x1 = (2+2+5+4)/4 = 3.25
x2 = (10+5+8+9)/4 = 8

New centroid: **c: (5.5, 3.75)**
h: (3.25, 8)

3b.

	Data objects (x1, x2)		Distance to c		Distance to h		Cluster
			5.5	3.75	3.25	8	
a	2	10	9.75		3.25		h
b	2	5	4.75		4.25		h
c	8	4	2.75		8.75		c
d	5	8	4.75		1.75		h
e	7	5	2.75		6.75		c
f	6	4	0.75		6.75		c
g	1	2	6.25		8.25		c
h	4	9	6.75		1.75		h

Memberships: **c cluster: c, e, f, g**
h cluster: a, b, d, h

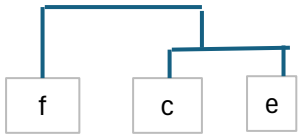
4a.

	a	b	c	d	e	f	g	h
a								
b	5							
c	8	6						
d	4	4	5					
e	7	5	1	4				
f	7	4	2	4	1			
g	8	3	7	7	7	5		
h	2	4	6	1	5	5	8	

cluster: c/e

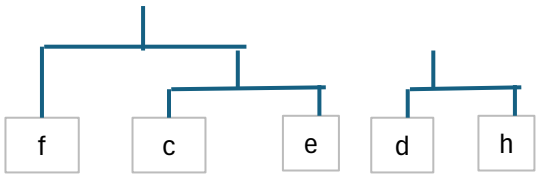
	c/e	a	b	d	f	g	h
c/e	0	7	5	4	1	7	5
a	7	0	5	4	7	8	2
b	5	5	0	4	4	3	4
d	4	4	4	0	4	7	1
f	1	7	4	4	0	5	5
g	7	8	3	7	5	0	8
h	5	2	4	1	5	8	0

cluster: c/e,f



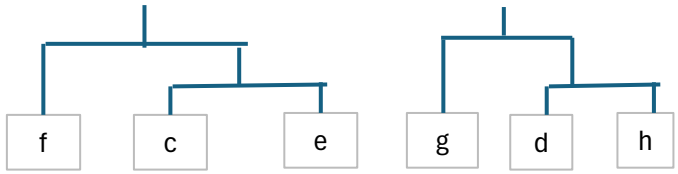
	c/e/f	a	b	d	g	h
c/e/f	0	7	4	4	5	5
a	7	0	5	4	8	2
b	4	5	0	4	3	4
d	4	4	4	0	7	1
g	5	8	3	7	0	8
h	5	2	4	1	8	0

cluster: c/e/f; d/h



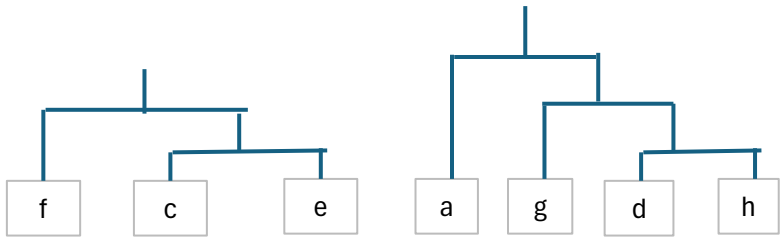
	c/e/f	a	b	d/h	g
c/e/f	0	7	4	4	5
a	7	0	5	2	8
b	4	5	0	4	3
d/h	4	2	4	0	1
g	5	8	3	7	0

cluster: c/e/f; d/h,g



	c/e/f	a	b	d/h/g
c/e/f	0	7	4	4
a	7	0	5	2
b	4	5	0	3
d/h/g	4	2	3	0

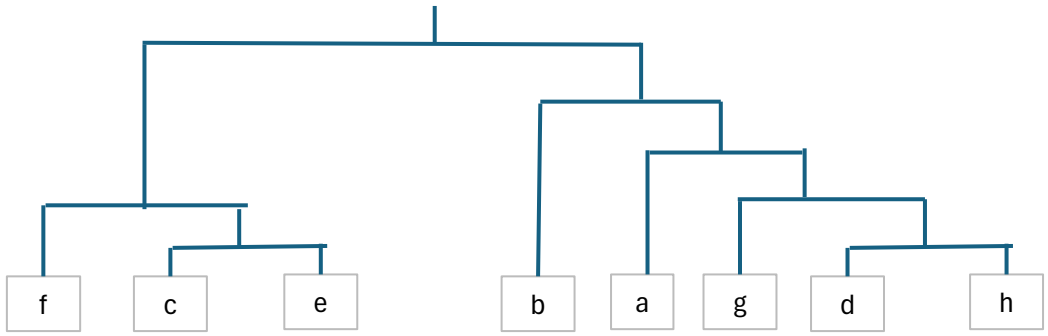
cluster: c/e/f; d/h/g,a



	c/e/f	b	d/h/g/a
c/e/f	0	4	4
b	4	0	3
d/h/g/a	4	3	0

cluster: c/e/f; d/h/g/a,b

Final dendogram:



4b.

Point aa distance: 0 < 6
Point ab distance: 5 < 6
Point ac distance: 8 > 6
Point ad distance: 4 < 6
Point ae distance: 7 > 6
Point af distance: 7 > 6
Point ag distance: 8 > 6
Point ah distance: 2 < 6
# of points within eps: 4 which is > minPts of 2 so a is a core point

Point ba distance: 5 < 6
Point bb distance: 0 < 6
Point bc distance: 6 = 6
Point bd distance: 4 < 6
Point be distance: 5 < 6
Point bf distance: 4 < 6
Point bg distance: 3 < 6
Point bh distance: 4 < 6
# of points within eps: 8 which is > minPts of 2 so be is a core point

Point ca distance: 8 > 6
Point cb distance: 6 = 6
Point cc distance: 0 < 6
Point cd distance: 5 < 6
Point ce distance: 1 < 6
Point cf distance: 2 < 6
Point cg distance: 7 > 6
Point ch distance: 6 = 6
# of points within eps: 6 which is > minPts of 2 so c is a core point

Point da distance: 4 < 6
Point db distance: 4 < 6
Point dc distance: 5 < 6
Point dd distance: 0 < 6
Point de distance: 4 < 6
Point df distance: 4 < 6
Point dg distance: 7 > 6
Point dh distance: 1 < 6
# of points within eps: 7 which is > minPts of 2 so d is a core point

Point ea distance: 7 > 6
Point eb distance: 5 < 6
Point ec distance: 1 < 6
Point ed distance: 4 < 6
Point ee distance: 0 < 6
Point ef distance: 1 < 6
Point eg distance: 7 > 6
Point eh distance: 5 < 6
# of points within eps: 6 which is > minPts of 2 so e is a core point

Point fa distance: 7 > 6
Point fb distance: 4 < 6
Point fc distance: 2 < 6
Point fd distance: 4 < 6
Point fe distance: 1 < 6
Point ff distance: 0 < 6
Point fg distance: 5 < 6
Point fh distance: 5 < 6
# of points within eps: 7 which is > minPts of 2 so f is a core point

Point ga distance: 8 > 6
Point gb distance: 3 < 6
Point gc distance: 7 > 6
Point gd distance: 7 > 6
Point ge distance: 7 > 6
Point gf distance: 5 < 6
Point gg distance: 0 < 6
Point gh distance: 8 > 6
# of points within eps: 3 which is > minPts of 2 so g is a core point

Point ha distance: 2 < 6
Point hb distance: 4 < 6
Point hc distance: 6 = 6
Point hd distance: 1 < 6
Point he distance: 5 < 6
Point hf distance: 5 < 6
Point hg distance: 8 > 6
Point hh distance: 0 < 6
# of points within eps: 7 which is > minPts of 2 so h is a core point

All points (a, b, c, d, e, f, g, h) are core points since they satisfy both parameters (eps=6 and minPts=2)