## EDA: ficha con la descripción de las pruebas para la corrección de la práctica $1\ (2016)$

tud de la localidad con las coordenadas creadas, escribelnfoGrados y escribeln- foGps.  P02.java Se crean diferentes localidades y se invoca setLatitud y setLongitud de todas ellas, de manera que todas las localidades tienen coordenadas asignadas. See crea un Atlas y se invoca setLocalidad con todas las localidades creadas, de manera que unas veces devuelve cierto y otras falso.  P03.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene algunas loca- lidades con coordenadas incorrectas.  P04.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada com- pletamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.  P05.java Se cinvoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  P06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas se invoca escribelnfoGrados y escribelnfoGps de todas las localidades.  P07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordena- das. Se crea una Atlas y se invoca getGps, comprobando que no tenga coordena- das. Se crea una Atlas y se invoca setLatitud, setLongitud de la localidade con ellas. Se invoca getGps, mostrando sus valores.  P08.java Se crea una Atlas y se invoca leeAtlas con un fichero que contiene localidades con coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  P09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le a	p01.java	Se crea una localidad (PLoc) y 2 coordenadas. Se invoca setLatitud y setLongi-	0.5
P02.java   Se crean diferentes localidades y se invoca setLatitud y setLongitud de todas ellas, de manera que todas las localidades tienen coordenadas asignadas. See crea un Atlas y se invoca setLocalidad con todas las localidades creadas, de manera que unas veces devuelve cierto y otras falso.    P03.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene algunas localidades con coordenadas incorrectas. Se crea una localidad, se le asignan coordenadas y se invoca muestra Atlas Parcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contiene una localidad.    P05.java   Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.    P06.java   Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribelnfoGrados y escribelnfoGps de todas las localidades.    P07.java   Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.    P08.java   Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.    P09.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestra Atlas Parcial de manera que muestra una submatriz situada com pletamente dentro de la matriz original en la que algunas de sus posiciones no con		tud de la localidad con las coordenadas creadas, escribe InfoGrados y escribe In-	
ellas, de manera que todas las localidades tienen coordenadas asignadas. See crea un Atlas y se invoca setLocalidad con todas las localidades creadas, de manera que unas veces devuelve cierto y otras falso.  p03.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene algunas localidades con coordenadas incorrectas.  p04.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.  p05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  p06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se c		foGps.	
crea un Atlas y se invoca setLocalidad con todas las localidades creadas, de manera que unas veces devuelve cierto y otras falso.  P03.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene algunas localidades con coordenadas incorrectas.  P04.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.  P05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  P06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  P07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  P08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  P09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  P10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas corre	p02.java	Se crean diferentes localidades y se invoca setLatitud y setLongitud de todas	0.5
manera que unas veces devuelve cierto y otras falso.  p03.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene algunas localidades con coordenadas incorrectas.  p04.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.  p05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crea dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidade, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con una de ellas y se invoca s		ellas, de manera que todas las localidades tienen coordenadas asignadas. See	
D03.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene algunas localidades con coordenadas incorrectas.   D04.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.   D05.java   Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.   D06.java   Se crean diferentes localidades, todas ellas con ciudad asignada.   D0.5 crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.   D0.5 crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crea das. Se crea una Set invoca getGps, comprobando que no tenga coordenadas. Se con conclenadas y se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.   D0.5 con coordenadas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.   D10.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas		crea un Atlas y se invoca setLocalidad con todas las localidades creadas, de	
D04.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.    D05.java   Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.    D06.java   Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.    D07.java   Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crea das. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.    D08.java   Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.    D09.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contiene una localidad.    D10.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		manera que unas veces devuelve cierto y otras falso.	
D04.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.    D05.java   Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada. Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.    D07.java   Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crea das. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.    D08.java   Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.    D09.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.    D10.java   Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con una de ellas	p03.java	Se crea un Atlas y se invoca leeAtlas con un fichero que contiene algunas loca-	0.5
con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.  P05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribelnfoGrados y escribelnfoGps de todas las localidades.  P07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se invoca getGps, mostrando sus valores.  P08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crea distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  P09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  P10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		lidades con coordenadas incorrectas.	
invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.  P05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  P06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  P07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenado con ellas. Se invoca getGps, mostrando sus valores.  P08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  P09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  P10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con	p04.java	Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades	0.5
pletamente dentro de la matriz original en la que todas sus posiciones contienen una localidad.  p05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  p06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crea una localidad y se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una fichero que contiene solo localidades con coordenadas correctas. Se crea una fichero que contiene solo localidades con coordenadas correctas. Se crea dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se	
p05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  p06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		invoca muestra Atlas Parcial de manera que muestra una submatriz situada com-	
p05.java Se invoca la aplicación Continente pasándole como parámetro un fichero que contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  p06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		pletamente dentro de la matriz original en la que todas sus posiciones contienen	
contiene solo localidades con coordenadas correctas, y en el que solo faltan por etiquetar unas pocas localidades, todas ellas con ciudad asignada.  P06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  P07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  P08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  P09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  P10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		una localidad.	
etiquetar unas pocas localidades, todas ellas con ciudad asignada.  p06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribelnfoGrados y escribelnfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con	p05.java	Se invoca la aplicación Continente pasándole como parámetro un fichero que	0.5
p06.java Se crean diferentes localidades de manera que a todas les falta algún dato. Se crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contiene una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		contiene solo localidades con coordenadas correctas, y en el que solo faltan por	
crean una serie de coordenadas y se invoca setLatitud y setLongitud de todas las localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.  p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		etiquetar unas pocas localidades, todas ellas con ciudad asignada.	
localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca escribeInfoGrados y escribeInfoGps de todas las localidades.    p07.java	p06.java	Se crean diferentes localidades de manera que a todas les falta algún dato. Se	0.5
p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		crean una serie de coordenadas y se invoca set Latitud y set Longitud de todas las	
p07.java Se crea una localidad y se invoca getGps, comprobando que no tenga coordenadas. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		localidades, de manera que todas tengan sus coordenadas asignadas. Se invoca	
das. Se crean dos coordenadas, se invoca setLatitud, setLongitud de la localidad con ellas. Se invoca getGps, mostrando sus valores.  P08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  P09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  P10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		escribeInfoGrados y escribeInfoGps de todas las localidades.	
con ellas. Se invoca getGps, mostrando sus valores.  P08.java  Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  P09.java  Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  P10.java  Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con	p07.java	Se crea una localidad y se invoca getGps, comprobando que no tenga coordena-	0.5
p08.java Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		das. Se crean dos coordenadas, se invoca set Latitud, set Longitud de la localidad	
con coordenadas correctas. Se crean distintas localidades a las que se les asigna coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		con ellas. Se invoca getGps, mostrando sus valores.	
coordenadas y se invoca consultaAtlas con todas estas localidades, que están situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con	p08.java	Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades	0.5
situadas en el atlas y tienen ciudad asignada.  p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		con coordenadas correctas. Se crean distintas localidades a las que se les asigna	
p09.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		coordenadas y se invoca consulta Atlas con todas estas localidades, que están	
con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y se invoca muestra Atlas Parcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java  Se crea un Atlas y se invoca lee Atlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca set Localidad, consulta Atlas y muestra Atlas Parcial con		situadas en el atlas y tienen ciudad asignada.	
se invoca muestraAtlasParcial de manera que muestra una submatriz situada completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con	p09.java	Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades	0.5
completamente dentro de la matriz original en la que algunas de sus posiciones no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y	
no contienen una localidad.  p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		se invoca muestra Atlas Parcial de manera que muestra una submatriz situada	
p10.java Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		completamente dentro de la matriz original en la que algunas de sus posiciones	
con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con		no contienen una localidad.	
una de ellas y se invoca setLocalidad, consultaAtlas y muestraAtlasParcial con	p10.java	Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades	0.5
		con coordenadas correctas. Se crean dos localidades, se le asignan coordenadas a	
ambas localidades.		una de ellas y se invoca set Localidad, consulta Atlas y muestra Atlas Parcial con	
		ambas localidades.	

D

p11.java	Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades	0.5
	con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y	
	se invoca muestraAtlasParcial de manera que muestra una submatriz situada	
	completamente dentro de la matriz original en la que algunas de sus posiciones	
	contienen una localidad que no tiene asignada la ciudad.	
p12.java	Se crea una Atlas y se invoca leeAtlas con un fichero que solo contiene localidades	0.5
	con coordenadas correctas. Se crean distintas localidades a las que se les asigna	
	coordenadas y se invoca consulta Atlas con todas estas localidades, de manera	
	algunas de ellas no están situadas en el atlas o no tienen ciudad asignada, y	
	otras sí.	
p13.java	Se crea un Atlas y se invoca leeAtlas con un fichero que contiene solo localidades	0.5
	con coordenadas correctas. Se crea una localidad, se le asignan coordenadas y	
	se invoca muestraAtlasParcial de manera que muestra una submatriz situada	
	parcialmente fuera de la matriz original en la que todas sus posiciones contienen	
	una localidad con la ciudad asignada.	
p14.java	Se invoca la aplicación Continente pasándole como parámetro un fichero que	0.5
	contiene solo las localidades correspondientes a los representantes, y algunos de	
	ellos no tienen ciudad asignada.	
p15.java	Se invoca la aplicación Continente pasándole como parámetro un fichero que	0.5
	contiene algunas localidades con coordenadas incorrectas, y en el que solo faltan	
	por etiquetar unas pocas localidades, todas ellas con ciudad asignada.	
p16.java	Se invoca la aplicación Continente pasándole como parámetro un fichero que	0.5
	contiene algunas localidades sin etiquetar con coordenadas repetidas que debe-	
	rían ser descartadas, y en el que solo faltan por etiquetar unas pocas localidades,	
	todas ellas con ciudad asignada.	
p17.java	Se invoca la aplicación Continente pasándole como parámetro un fichero que	0.5
	contiene solo localidades con coordenadas correctas, y en el que faltan por eti-	
	quetar más de la mitad de las localidades, todas ellas con ciudad asignada.	
p18.java	Se invoca la aplicación Continente pasándole como parámetro un fichero que	0.5
	contiene solo localidades con coordenadas correctas, y en el que faltan por eti-	
	quetar más de la mitad de las localidades, algunas de ellas sin ciudad asignada.	