```
from graph import Graph

def graph_from_edgelist(E, directed=False):
  """Make a graph instance based on a sequence of edge tuples.

  Edges can be either of from (origin,destination) or
  (origin,destination,element). Vertex set is presume to be those
  incident to at least one edge.

  vertex labels are assumed to be hashable.
  """
  g = Graph(directed)
  V = set()
  for e in E:
    V.add(e[0])
    V.add(e[1])

  verts = {}  # map from vertex label to Vertex instance
  for v in V:
    verts[v] = g.insert_vertex(v)

  for e in E:
    src = e[0]
    dest = e[1]
    element = e[2] if len(e) > 2 else None
    g.insert_edge(verts[src],verts[dest],element)

  return g

def figure_14_3():
  """Return the unweighted, directed graph from Figure 14.3 of DSAP."""
  E = (
    ('BOS','SFO'), ('BOS','JFK'), ('BOS','MIA'), ('JFK','BOS'),
    ('JFK','DFW'), ('JFK','MIA'), ('JFK','SFO'), ('ORD','DFW'),
    ('ORD','MIA'), ('LAX','ORD'), ('DFW','SFO'), ('DFW','ORD'),
    ('DFW','LAX'), ('MIA','DFW'), ('MIA','LAX'),
    )
  return graph_from_edgelist(E, True)

def figure_14_8():
  """Return the unweighted, directed graph from Figure 14.8 of DSAP."""
  E = (
    ('BOS','SFO'), ('BOS','JFK'), ('BOS','MIA'), ('JFK','BOS'),
    ('JFK','DFW'), ('JFK','MIA'), ('JFK','SFO'), ('ORD','DFW'),
    ('ORD','MIA'), ('LAX','ORD'), ('DFW','SFO'), ('DFW','ORD'),
    ('DFW','LAX'), ('MIA','DFW'), ('MIA','LAX'), ('SFO','LAX'),
    )
  return graph_from_edgelist(E, True)

def figure_14_9():
  """Return the unweighted, undirected graph from Figure 14.9 of DSAP.

  This is the same graph as in Figure 14.10.
  """
  E = (
    ('A','B'), ('A','E'), ('A','F'), ('B','C'), ('B','F'),
    ('C','D'), ('C','G'), ('D','G'), ('D','H'), ('E','F'),
    ('E','I'), ('F' 'I'), ('G','J'), ('G','K'), ('G','L'),
    ('H','L'), ('I','J'), ('I','M'), ('I','N'), ('J','K'),
    ('K','N'), ('K','O'), ('L','P'), ('M','N'),
    )
  return graph_from_edgelist(E, False)

figure_14_10 = figure_14_9   # same graph

def figure_14_11():
  """Return the unweighted, directed graph from Figure 14.11 of DSAP."""
  E = (
    ('BOS','JFK'), ('BOS','MIA'), ('JFK','BOS'), ('JFK','DFW'),
    ('JFK','MIA'), ('JFK','SFO'), ('ORD','DFW'),
    ('LAX','ORD'), ('DFW','SFO'), ('DFW','ORD'),
    ('DFW','LAX'), ('MIA','DFW'), ('MIA','LAX'),
    )
  return graph_from_edgelist(E, True)

def figure_14_12():
  """Return the unweighted, directed graph from Figure 14.12 of DSAP.

  This is the same graph as Figure 14.13.
  """
  E = (
    ('A','C'), ('A','D'), ('B','D'), ('B', 'F'), ('C','D'), ('C','E'),
    ('C','H'), ('D','F'), ('E','G'), ('F','G'), ('F','H'), ('G','H')
    )
  return graph_from_edgelist(E, True)

figure_14_13 = figure_14_12   # same graph

def figure_14_14():
  """Return the weighted, undirected graph from Figure 14.14 of DSAP."""
  E = (
    ('SFO', 'LAX', 337), ('SFO', 'BOS', 2704), ('SFO', 'ORD', 1846),
    ('SFO', 'DFW', 1464), ('LAX', 'DFW', 1235), ('LAX', 'MIA', 2342),
    ('DFW', 'ORD', 802), ('DFW', 'MIA', 1121), ('ORD', 'BOS', 867),
    ('ORD', 'JFK', 740), ('MIA', 'JFK', 1090), ('MIA', 'BOS', 1258),
    ('JFK', 'BOS', 187),
    )
  return graph_from_edgelist(E, False)

def figure_14_15():
  """Return the weighted, undirected graph from Figure 14.15 of DSAP.

  This is the same graph as in Figures 14.16, 14.17, and and 14.20-14.24.
  """
  E = (
    ('SFO', 'LAX', 337), ('SFO', 'BOS', 2704), ('SFO', 'ORD', 1846),
    ('SFO', 'DFW', 1464), ('LAX', 'DFW', 1235), ('LAX', 'MIA', 2342),
    ('DFW', 'ORD', 802), ('DFW', 'JFK', 1391), ('DFW', 'MIA', 1121),
    ('ORD', 'BOS', 867), ('ORD', 'PVD', 849), ('ORD', 'JFK', 740),
    ('ORD', 'BWI', 621), ('MIA', 'BWI', 946), ('MIA', 'JFK', 1090),
    ('MIA', 'BOS', 1258), ('BWI', 'JFK', 184), ('JFK', 'PVD', 144),
    ('JFK', 'BOS', 187),
    )
    return graph_from_edgelist(E, False)
```