Ski-O

- name directories and files as directed
  - directory – `skio`
  - file – `SkiO.java`
  - resubmit directory – `skiov2`

- graph structure – handling of controls
  - need vertices for both junctions and controls
  - controls should be treated as being in the middle of a track segment
    - split the edges for that track segment – add (from,control), (control,to) with half the length of (from,to) for each direction

Ski-O

- for flexibility, prefer the most general class that is appropriate
  - use the concrete class (`Adjacency*DirectedGraph`, `ArrayList`, `HashMap`, etc) only when you have to
    - (for creating objects)
  - otherwise use `DirectedGraph`, `Vertex`, `Edge`, `Map`, `List`, ...

- efficiency
  - choose the right graph representation
    - what operations does Dijkstra's algorithm use – is adjacency list or adjacency matrix better for those?
  - it's inefficient to search when it's a lookup task...