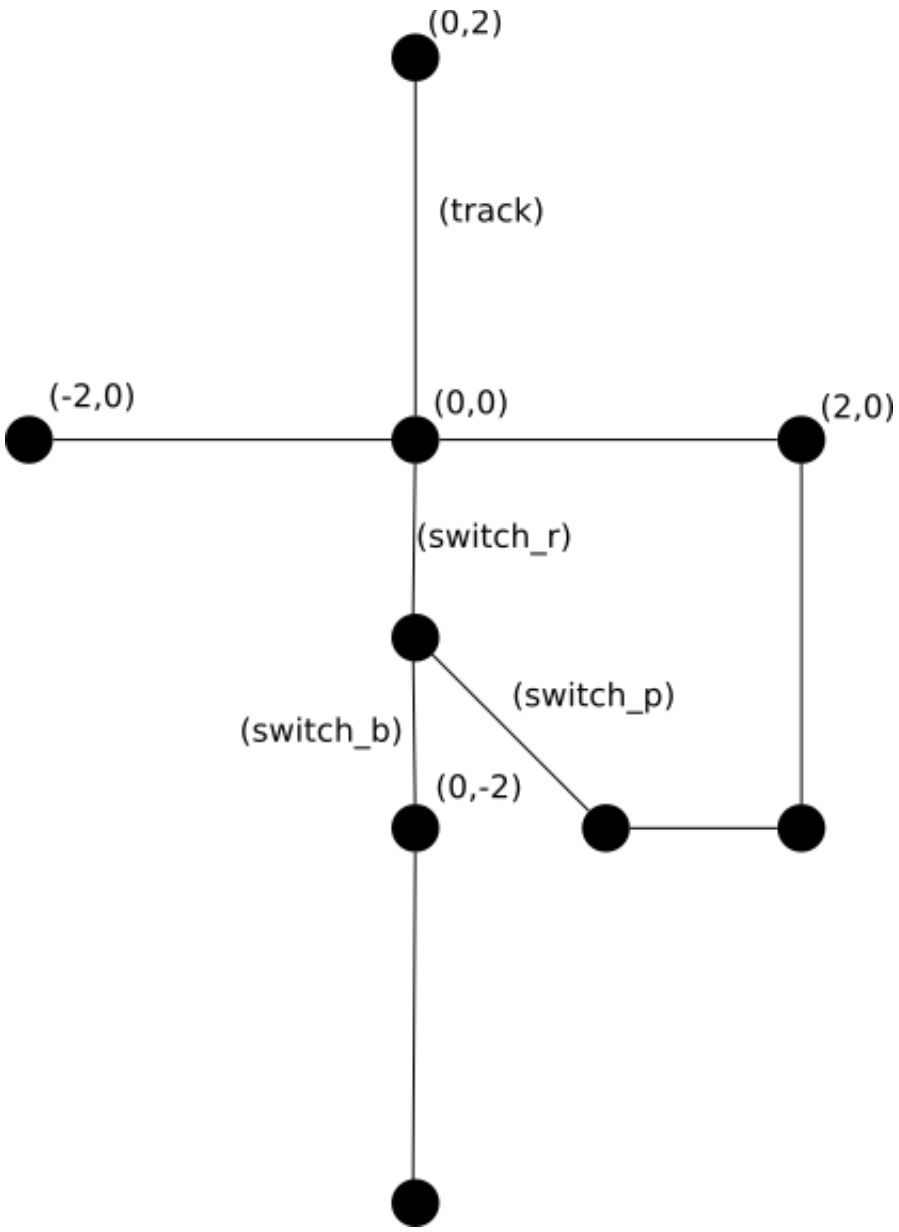
Team 4A
SQL Track Representation

In the interest of providing a table of the track which is both entirely described and easy to implement, teams 4 A/B have decided to design a zero-mass graph in the form of a single-linked list. The current testing table is described below.

| **Name:** | **id** | **from** | **x** | **y** | **track_type** | **ds** |
|---|---|---|---|---|---|---|
| Type: | Integer, unique key. | Previous section's id. Integer. | First quadrant x value, inches. Float. | First quadrant y value, inches. Float. | Enum: track, switch_r, switch_p, switch_b | ID of detection section, reference from separate table. Integer. |

This format allows for a reasonably small table which can describe the entire track by assuming an uncomplex graph type where all edges have associated vertices. Each piece of track or branch of a switch defines a destination x,y tuple and has an origin associated with the coordinates of the 'from' section behind it to create one edge bounded by two vertices.

The table currently holds a sample of imaginary track from the following template. Table contents have not been updated yet to reflect the 1'st quadrant coordinate decision, in which all values are to be associated with space in the first quadrant of a Cartesian plane to avoid handling negative values in computation.


(0,2)
(track)
(-2,0)
(0,0)
(2,0)
(switch_r)
(switch_p)
(switch_b)
(0,-2)