1 passo:

Passar o vetor de 2x, luma coordenadas recebido da API crows come le Loutra com 0 revier no algoritmo já Concatera todos exclus 100 implementado deliterer mol ubso realidadão 1º chamada: coster original A + B AB repelides: Concatema axchei 0 740w

300m

Passar o mesmo vetor, porém invertido (cuidado para não inverter latitude com longitude) no algoritmo já implementado.

3 passo: Usar função truncate (turfe.js) para precisar as casas decimais das coordenadas de cada posição do vetor.

> var truncated = turf.truncate(point, {precision: 6, coordinates: 2}); //=truncated.geometry.coordinates

4 passo:

Usar função para excluir coordenadas duplicadas consecutivas do vetor

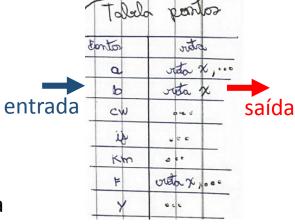
5 passo:

2 passo:

umroodin

2º dramada:

Pesquisar cada ponto na tabela de pontos e verificar se ele pertence a alguma rota



- 6 passo: Buscar os points (coordenadas) de cada Rota encontrada na Tabela de Pontos (que serão as rotas que apresentam pontos em comum com a rota a ser validada). Poderá haver mais de 1 rota em conflito. Fazer o cálculo para TODAS AS ROTAS.
- **7 passo:** Chamar a função lineOverlap passando a rota a ser validada e a rota retirada da tabela de pontos.

```
var line1 = turf.lineString([[115, -35], [125, -30], [135, -30], [145, -35]]); //rota x ou rotaConflitante
var line2 = turf.lineString([[115, -25], [125, -30], [135, -30], [145, -25]]); // rota a ser validada
var overlapping = turf.lineOverlap(line1, line2, {tolerance: 0.005});
```

- 8 passo: Calcular distância entre coordenadas que foram retornadas pelo lineOverlap
- **9 passo:** Invalidar a rota a ser cadastrada caso a distância seja maior que um valor pré estabelecido
- **10 passo:** Exibir as várias retornadas pelo código no mapa caso a rota é dada como inválida. Exibir mensagem de Cadastrado com Sucesso caso não seja encontrado nenhum trecho conflitante em nenhuma rota buscada na tabela de pontos (variável trechosComOverlap retorna null para todas as rotasConflitantes pesquisadas.)