



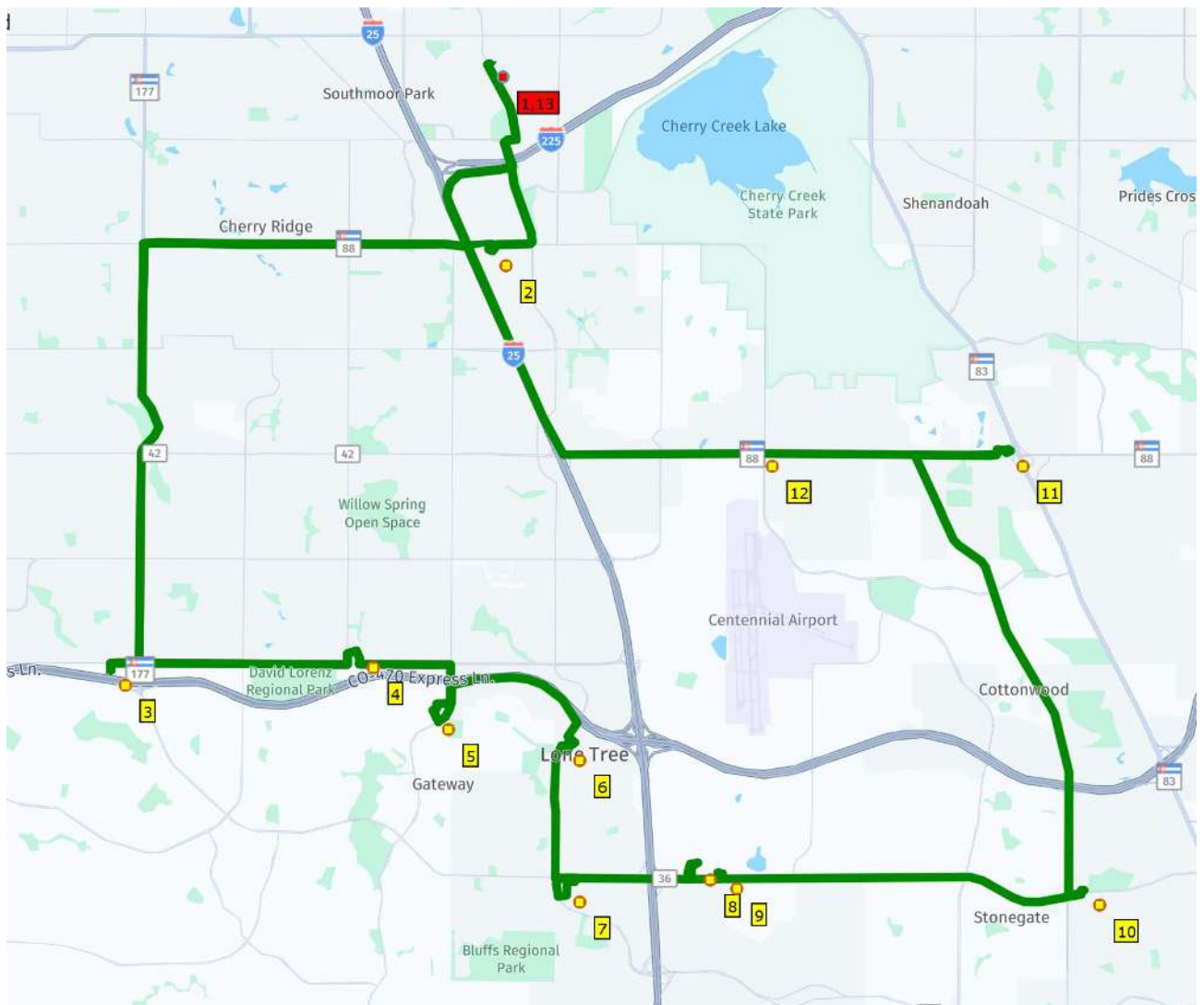
# RouteSavvy Route Optimization API Documentation

## Introduction:

For developers seeking an affordable, powerful route optimization engine, the [RouteSavvy API](#) routing engine is one of the most robust options on the market today. This versatile, flexible routing engine is a REST API, making it an easy-to-use, cost-effective service. The RouteSavvy routing engine offers a wealth of options and benefits for developers, including:

- An affordable, scalable pricing structure
- Support for multiple programming languages
- Two route type options to choose from
- Free code samples...and more

The RouteSavvy API is REST-based service for optimizing driving stop order for a single vehicle route with multiple stops. The RouteSavvy API supports GET and POST requests and returns results in JSON format for optimal use in web applications. The optimized results return stops ordered for the most efficient driving route. Results include turn-by-turn directions for each route leg and a latitude-longitude polyline describing the route for use in map visualization. Onterra Systems offers a [RouteSavvy API Free Trial](#) to get started with evaluating the service.



RouteSavvy API: This is a sample of optimized stops and route path displayed on a map.

## API Endpoint:

- Base URL: <https://optimizer2.routesavvy.com>
- Service wsdl URL: <https://optimizer2.routesavvy.com/RSAPI.svc>
- GET URL: <https://optimizer2.routesavvy.com/RSAPI.svc/GetOptimize?query={jsonquery}>
- POST URL: <https://optimizer2.routesavvy.com/RSAPI.svc/PostOptimize>

### Note:

GET requests (RSAPI.svc/GetOptimize) requests are limited to the URL encoded parameter maximum length. Although this varies by browser, 2,048 characters is the typical suggested limit for encoding stops in the URL string. Because of the URL length maximum, you will be limited to a smaller number of customer stops that can be optimized using this approach. POST requests (RSAPI.svc/PostOptimize) requests do not have URL length limitations and can accommodate larger numbers of customer stops.

## API Request Input Parameters:

There are two primary components of the API request, *Locations* array and *OptimizeParameters*. The *Locations* array are the input details about the customer stop locations and the *OptimizeParameters* are the input details about the vehicles that will visit the customer stop locations.

### Locations:

- **Name:**  
This is the custom name you can specify for the customer stop locations. If no Name is defined for a stop location, results will identify "Name" as *null* and the location can only be identified by its Latitude/Longitude.
- **Latitude/Longitude:**  
Latitude/longitude for the specified customer stop locations and the start and end locations for the definition of vehicle/driver parameters. The first location is the starting location and the last location is the ending location. If you wish to optimize a round-trip route, duplicate the starting location as the ending location.

RouteSavvy API utilizes a fixed-start and fixed-end optimization. The definition of these two parameters can have a considerable impact on the optimization results. RouteSavvy API does not support fixed-start and open-end optimization scenarios. As a result, it is critical to always consider the most appropriate definition of starting and ending locations for best optimization results.

6-decimal latitude/longitude accuracy is recommended to get the best optimization results. Lower accuracy latitude/longitude data may be used but may cause an error in API processing as well as inaccurate or unusable route results. If you do not have latitude/longitude data and only have street addresses, any address geocoding service can be used to convert street address data to latitude/longitude.

OnTerra Systems can provide access to commercial geocoding services if needed. Please contact [info@routesavvy.com](mailto:info@routesavvy.com) for licensing and pricing information related to available geocoding services.

- **VisitDurationInMinutes:**  
The API also incorporates expected stop duration or the amount of time expected to complete each stop, defined in the *VisitDurationInMinutes* parameter. Visit duration specifies how long a visit, such as delivery, pick-up or repair may take at the customer stop location. If no stop duration is defined, then default of 0 minutes will be assumed. *VisitDurationInMinutes* is optional but is recommended if scheduling is critical to optimization needs. Please note, a stop may be assigned to

a specific time window as long as the resultant beginning time of the stop falls within that time window. It is possible that the resultant ending time of the stop will occur after the close of the time window. The optimization only considers that the stop begins within the assigned time frame, and the stop duration may run after the close of the time window. For vehicle start and end locations, *VisitDurationInMinutes* is not recommended since it typically doesn't apply to such locations.

### **OptimizeParameters:**

- **AppID:**

AppID is your credentials (alphanumeric token) for accessing the API. If you are a current user of our legacy RouteSavvy API, your existing AppID can be used for trial/dev/test access to RouteSavvy API Plus. If you do not have credentials, you can register for a free trial and receive credentials any time at the following link: <https://www.routesavvy.com/routesavvy-api-free-trial/>

- **OptimizeType:**

*OptimizeType* specifies whether “time” or “distance” should be used to calculate the travel optimization. The “time” option will produce the optimization result with the shortest drive time, while “distance” will produce the result based purely on the shortest driving distance.

The routing used is based on standard vehicle driving directions. If more advanced routing parameters are required, such as the ability to avoid certain commercial vehicle or truck routing restrictions, such as hazmat or vehicle dimension (height, width, weight) restrictions, OnTerra Systems also offers this through our RouteSavvy API Plus Premium. Please contact [info@routesavvy.com](mailto:info@routesavvy.com) for more details including information on licensing and pricing for advanced routing capabilities.

- **RouteType:**

*RouteType* supports ‘basic’ and ‘realroadcar’ as possible input types. ‘realroadcar’ calculates the route based on the road network and supports up to 300 stops. ‘basic’ calculates the route based on great circle distance (not the road network) and supports up to 500 stops.

- **Avoid:**

*Avoid* supports road types ‘tolls’ or ‘highways’ as road types that you can specify be avoided for the route. If the route doesn't have any requirements to avoid these road types you can use input ‘none’.

- **Departure:**

*Departure* is the date and time that the vehicle will depart for the route and must be specified. The departure value must be in the future in the date-time format *yyyy-MM-ddTHH:mm:ss*. For example, if *Departure=2019-01-30T08:00:00* and the origin point is 39.595140, -104.906670 (Colorado, USA), this will indicate a departure at 8:00 AM Mountain Standard Time (MST) or 2019-01-30T15:00:00+00:00 UTC. The default departure behavior can be overridden by adding a time zone offset (*yyyy-MM-ddTHH:mm:ss-HH:00*) to the departure datetime.

For example:

Request:

“Departure”: “2020-01-23T08:00:00-08:00” will force time to PST and override time zone of origin location.

Result:

“Arrival”: “2020-01-23T16:11:56+00:00”,

“Departure”: “2020-01-23T16:16:56+00:00”

RouteTypes ‘basic’ and ‘realroadcar’ only use the departure time for labeling directions and calculating stop arrival/departure times but does not account for traffic variations. The depart date is always today’s date and time zone offsets are ignored.

**API Request Input Parameter Details:**

Parameter Name	Data Type	Required?	Description	Example
Locations array: <i>Name</i>	string	No	Custom name value for the customer stop location.	“Customer01”
Locations array: <i>Latitude</i>	float	Yes	Latitude value for the stop location and the vehicle start and end locations.	“39.595140”
Locations array: <i>Longitude</i>	float	Yes	Longitude value for the stop location and the vehicle start and end locations.	“-104.849620”
Locations array: <i>VisitDurationInMinutes</i>	integer	No	Expected visit duration for the stop location, in minutes.	“10”
OptimizeParameters: <i>AppID</i>	String	Yes	Your Onterra AppID value to authenticate the API request.	“90c13....21a609”
OptimizeParameters: <i>OptimizeType</i>	string	No	Specify whether the route optimization should be based on time or distance. Valid input values are “Time” or “Distance”. If not specified, “Time” is the default.	“Time”
OptimizeParameters: <i>RouteType</i>	string	No	Specify whether to use route type ‘basic’ or ‘realroadcar’. If not specified, “basic” is the default.	“realroadcar”
OptimizeParameters: <i>Avoid</i>	string	No	Specify to avoid road type ‘tolls’, ‘highways’ or not to avoid those road types (‘none’). If not specified, “none” is the default.	“none”
OptimizeParameters: <i>Departure</i>	string	Yes	Specify the date and time of the vehicle’s route departure. The departure value must	“2020-05 23T17:30:00-08:00”

			be in the future in the date-time format: yyyy-MM-ddTHH:mm:ss	
--	--	--	--	--

### API Response Output Fields:

The JSON response from the API request includes the route stop details for the optimized route that was generated. Below is a table that has output field details, descriptions and examples returned in the API response.

#### API Response Output Field Details:

Field Name	Data Type	Description	Example
<i>Message</i>	string	Status of the API response, such as "Success" if successful or an error detailed why the request failed.	"Success"
<i>OptimizedStops</i>		Includes the following info for the Vehicle_id: <ul style="list-style-type: none"> <li>• Arrival</li> <li>• Departure</li> <li>• Distance</li> <li>• Duration</li> <li>• IsDuplicate</li> <li>• Name</li> <li>• RouteLocation</li> <li>• StopTimeMinutes</li> </ul>	
<i>OptimizedStops: Arrival</i>	string	Date and time of arrival	"17:05:49 5/15/2023"
<i>OptimizedStops: Departure</i>	string	Date and time of departure	"17:10:49 5/15/2023"
<i>OptimizedStops: Distance</i>	string	Drive distance in miles and kilometers	"3.30 miles (5.31 km)"
<i>OptimizedStops: Duration</i>	string	Drive time in minutes and seconds	"5 minutes, 49 seconds"
<i>OptimizedStops: IsDuplicate</i>	boolean	Indicates if a specific stop location has been used more than once in a route.	false
<i>OptimizedStops: Name</i>	string	Name value for the stop location.	"Customer008"
<i>OptimizedStops: RouteLocation</i>	float	Latitude and longitude of the stop.	"Latitude": 39.59503 "Longitude": -104.80432
<i>OptimizedStops: StopTimeMinutes</i>	integer	Duration of the stop in minutes.	5
<i>Route</i>		Includes the following: <ul style="list-style-type: none"> <li>• DrivingDistance</li> <li>• DriveDistanceUnit</li> <li>• DriveTime</li> <li>• DriveTimeUnit</li> <li>• RouteLegs</li> <li>• RoutePath</li> </ul>	
<i>Route: DriveDistance</i>	float	Total driving distance of the route, in kilometers.	53.088826

Route: <i>DriveDistanceUnit</i>	string	Unit of measurement of the driving distance.	"Kilometer"
Route: <i>DriveTime</i>	integer	Total driving time of the route, in seconds.	3133
Route: <i>DriveTimeUnit</i>	string	Unit of measurement of the driving time.	"Second"
Route: <i>RouteLegs</i>		Includes the following: <ul style="list-style-type: none"> <li>• Directions</li> <li>• DriveDistance</li> <li>• DriveTime</li> <li>• LegBegin</li> <li>• LegEnd</li> </ul>	
Route: <i>RoutePath</i>	Array of floats	An array of latitudes and longitudes that make up the polyline of route path. This can be used to display the route path on a map.	[39.595137, -104.84962], [39.595134, -104.850088], [39.59559, -104.85009]
RouteLegs: <i>Directions</i>	Array of strings	Step by step driving instructions for the route leg.	"Make a U-turn onto South Holly Street", "Turn left onto East County Line Road", "Turn right onto South Quebec Street"
RouteLegs: <i>DriveDistance</i>	float	Driving distance of the route leg, in kilometers.	2.72656
RouteLegs: <i>DriveTime</i>	integer	Driving duration of the route leg, in seconds.	302
RouteLegs: <i>LegBegin</i>	Array of a string and a float	Name value of stop location that the leg begins at and the RouteLocation (latitude/longitude) of that stop location.	"Name": "Customer101", "RouteLocation": { "Latitude": 39.567508, "Longitude": -104.923069}
RouteLegs: <i>LegEnd</i>	Array of a string and a float	Name value of stop location that the leg ends at and the RouteLocation (latitude/longitude) of that stop location.	"Name": "Customer102", "RouteLocation": { "Latitude": 39.558012, "Longitude": -104.90667}
<i>UnassignedStops</i>	Array of strings	List of the stops that couldn't be assigned due to time constraints and vehicle availability, if applicable. It will include detailed info such as name, code, ID and reason.	"Name": "Customer001", "code": "2", "id": "location-1", "reason": "cannot be visited within time window"

### Technical Support:

For all support inquiries, please email [support@routesavvy.com](mailto:support@routesavvy.com) and be sure to mention that you are using the RouteSavvy API and include as many details as possible regarding your inquiry. We're happy to assist you with any questions or issues you may have.

### Example GET API request:

<https://www.routesavvy.com/routesavvy-route-optimization-api/>

Onterra Systems, LLC ©2023

The following POSTOptimize example includes 5 route stops with varying stop durations.

```
https://optimizer2.routesavvy.com/RSAPI.svc/GETOptimize?query={"Locations": [{"Name": "Customer A", "Latitude": 39.595140, "Longitude": -104.849620, "VisitDurationInMinutes": 10}, {"Name": "Customer B", "Latitude": 39.558012, "Longitude": -104.906670, "VisitDurationInMinutes": 5}, {"Name": "Customer C", "Latitude": 39.566128, "Longitude": -104.965901, "VisitDurationInMinutes": 10}, {"Name": "Customer D", "Latitude": 39.535220, "Longitude": -104.882080, "VisitDurationInMinutes": 10}, {"Name": "Customer E", "Latitude": 39.555669, "Longitude": -104.882878, "VisitDurationInMinutes": 5}], "OptimizeParameters": {"Appld": "<InsertYourTokenHere>", "OptimizeType": "distance", "RouteType": "realroadcar", "Avoid": "none", "Departure": "2020-05-23T17:30:00"}}
```

### Example POST API request:

The following POSTOptimize example includes 5 route stops with varying stop durations.

```
https://optimizer2.routesavvy.com/RSAPI.svc/PostOptimize
```

```
{
  "Locations": [
    {
      "Name": "Customer A",
      "Latitude": 39.595140,
      "Longitude": -104.849620,
      "VisitDurationInMinutes": 10
    },
    {
      "Name": "Customer B",
      "Latitude": 39.558012,
      "Longitude": -104.906670,
      "VisitDurationInMinutes": 5
    },
    {
      "Name": "Customer C",
      "Latitude": 39.566128,
      "Longitude": -104.965901,
      "VisitDurationInMinutes": 10
    },
    {
      "Name": "Customer D",
      "Latitude": 39.535220,
      "Longitude": -104.882080,
      "VisitDurationInMinutes": 10
    },
    {

```

<https://www.routesavvy.com/routesavvy-route-optimization-api/>

Onterra Systems, LLC ©2023



```
"Name": "Customer E",
  "Latitude": 39.555669,
  "Longitude": -104.882878,
  "VisitDurationInMinutes": 5
}
],
"OptimizeParameters":{
  "AppId": "<InsertYourTokenHere>",
  "OptimizeType": "distance",
  "RouteType": "realroadcar",
  "Avoid": "none",
  "Departure": "2020-05-23T17:30:00-08:00"
}
}
```

### Example API Response:

The following shows the API result from the above example GET and POST request.

```
{
  "Message": "Success",
  "OptimizedStops": [
    {
      "Arrival": null,
      "Departure": "17:30:00 11/28/2024",
      "Distance": null,
      "Duration": null,
      "IsDuplicate": false,
      "Name": "Customer A",
      "RouteLocation": {
        "Latitude": 39.59514,
        "Longitude": -104.84962
      },
      "StopTimeMinutes": 10
    },
    {
      "Arrival": "17:43:09 11/28/2024",
      "Departure": "17:53:09 11/28/2024",
      "Distance": "10.25 miles (16.50 km)",

```

```
"Duration": "13 minutes, 9 seconds",
"IsDuplicate": false,
"Name": "Customer C",
"RouteLocation": {
  "Latitude": 39.566128,
  "Longitude": -104.965901
},
"StopTimeMinutes": 10
},
{
  "Arrival": "17:58:57 11/28/2024",
  "Departure": "18:03:57 11/28/2024",
  "Distance": "4.03 miles (6.48 km)",
  "Duration": "5 minutes, 47 seconds",
  "IsDuplicate": false,
  "Name": "Customer B",
  "RouteLocation": {
    "Latitude": 39.558012,
    "Longitude": -104.90667
  },
  "StopTimeMinutes": 5
},
{
  "Arrival": "18:10:28 11/28/2024",
  "Departure": "18:20:28 11/28/2024",
  "Distance": "3.20 miles (5.16 km)",
  "Duration": "6 minutes, 30 seconds",
  "IsDuplicate": false,
  "Name": "Customer D",
  "RouteLocation": {
    "Latitude": 39.53522,
    "Longitude": -104.88208
  },
  "StopTimeMinutes": 10
},
{
  "Arrival": "18:24:28 11/28/2024",
  "Departure": "18:29:28 11/28/2024",
  "Distance": "1.67 miles (2.69 km)",
```

```
"Duration": "4 minutes, 0 seconds",
"IsDuplicate": false,
"Name": "Customer E",
"RouteLocation": {
  "Latitude": 39.555669,
  "Longitude": -104.882878
},
"StopTimeMinutes": 5
}
],
"Route": {
  "DriveDistance": 30.827243,
  "DriveDistanceUnit": "Kilometer",
  "DriveTime": 1768,
  "DriveTimeUnit": "Second",
  "RouteLegs": [
    {
      "Directions": [
        "Continue onto East Arapahoe Road",
        "Keep right and take I 25 South toward Colorado Springs",
        "Keep right and take E-470 North, CO 470 West toward Limon, Grand
Junction",
        "Keep right and take CO 470 West toward Grand Junction",
        "Keep right and take CO 470 West toward Grand Junction, Quebec Street",
        "Keep right and take CO 177 toward University Boulevard",
        "Keep right and take CO 177 North toward University Boulevard North",
        "Turn left onto East County Line Road",
        "Turn slight right onto East County Line Road"
      ],
      "DriveDistance": 16.496856,
      "DriveTime": 790,
      "LegBegin": {
        "Name": "Customer A",
        "RouteLocation": {
          "Latitude": 39.59514,
          "Longitude": -104.84962
        }
      },
      "LegEnd": {
```

```
"Name": "Customer C",
"RouteLocation": {
  "Latitude": 39.566128,
  "Longitude": -104.965901
}
},
{
  "Directions": [
    "Make a U-turn onto East County Line Road",
    "Turn right onto South University Boulevard",
    "Turn left toward CO 470 East",
    "Keep right and drive toward Quebec Street",
    "Keep right and drive toward Quebec Street South",
    "Turn right"
  ],
  "DriveDistance": 6.4789040000000009,
  "DriveTime": 348,
  "LegBegin": {
    "Name": "Customer C",
    "RouteLocation": {
      "Latitude": 39.566128,
      "Longitude": -104.965901
    }
  },
  "LegEnd": {
    "Name": "Customer B",
    "RouteLocation": {
      "Latitude": 39.558012,
      "Longitude": -104.90667
    }
  }
},
{
  "Directions": [
    "Make a U-turn",
    "Turn right onto South Quebec Street",
    "Turn slight right onto South Quebec Street",
    "Turn left onto East Lincoln Avenue",
```

```
    "Turn right onto Commons Street"
  ],
  "DriveDistance": 5.156595,
  "DriveTime": 391,
  "LegBegin": {
    "Name": "Customer B",
    "RouteLocation": {
      "Latitude": 39.558012,
      "Longitude": -104.90667
    }
  },
  "LegEnd": {
    "Name": "Customer D",
    "RouteLocation": {
      "Latitude": 39.53522,
      "Longitude": -104.88208
    }
  }
},
{
  "Directions": [
    "Make a U-turn onto Commons Street",
    "Keep left onto Commons Street",
    "Turn left onto East Lincoln Avenue",
    "Turn right onto South Yosemite Street",
    "Turn right onto Park Meadows Drive"
  ],
  "DriveDistance": 2.6948869999999996,
  "DriveTime": 240,
  "LegBegin": {
    "Name": "Customer D",
    "RouteLocation": {
      "Latitude": 39.53522,
      "Longitude": -104.88208
    }
  },
  "LegEnd": {
    "Name": "Customer E",
    "RouteLocation": {
```

```
"Latitude": 39.555669,  
"Longitude": -104.882878  
}  
}  
},  
"RoutePath": [  
  [  
    39.59514,  
    -104.84962  
  ],  
  [  
    39.595134,  
    -104.85079  
  ],  
  [  
    39.595134,  
    -104.855337  
  ],  
  [  
    39.595129,  
    -104.860039  
  ],  
  [  
    39.595117,  
    -104.865383  
  ],  
  [  
    39.595129,  
    -104.86553  
  ],  
  [  
    39.595127,  
    -104.868269  
  ],  
  [  
    39.595114,  
    -104.868782  
  ],  
]
```

```
[  
  39.595107,  
  -104.873234  
],  
[  
  39.59511,  
  -104.876173  
],  
[  
  39.595103,  
  -104.878565  
],  
[  
  39.595115,  
  -104.879794  
],  
[  
  39.595133,  
  -104.880789  
],  
[  
  39.595112,  
  -104.881418  
],  
[  
  39.595074,  
  -104.881909  
],  
[  
  39.595006,  
  -104.882537  
],  
[  
  39.594832,  
  -104.883533  
],  
[  
  39.594782,  
  -104.883882
```

```
],  
[  
  39.594762,  
  -104.884213  
],  
[  
  39.594905,  
  -104.885901  
],  
[  
  39.59494,  
  -104.886021  
],  
[  
  39.594978,  
  -104.886086  
],  
[  
  39.595062,  
  -104.886173  
],  
[  
  39.595109,  
  -104.886204  
],  
[  
  39.595215,  
  -104.88623  
],  
[  
  39.595329,  
  -104.886218  
],  
[  
  39.595388,  
  -104.886188  
],  
[  
  39.595441,
```



```
-104.886148  
],  
[  
  39.595488,  
  -104.886097  
],  
[  
  39.59554,  
  -104.885994  
],  
[  
  39.595571,  
  -104.885835  
],  
[  
  39.595569,  
  -104.885746  
],  
[  
  39.595553,  
  -104.885659  
],  
[  
  39.59553,  
  -104.885588  
],  
[  
  39.595504,  
  -104.885538  
],  
[  
  39.595412,  
  -104.885438  
],  
[  
  39.594367,  
  -104.884668  
],  
[
```

```
39.593079,  
-104.883954  
],  
[  
39.591481,  
-104.883097  
],  
[  
39.59029,  
-104.882443  
],  
[  
39.584826,  
-104.879294  
],  
[  
39.580761,  
-104.876942  
],  
[  
39.57903,  
-104.875947  
],  
[  
39.578499,  
-104.875678  
],  
[  
39.577976,  
-104.875448  
],  
[  
39.57749,  
-104.875271  
],  
[  
39.574468,  
-104.87433  
],
```

```
[  
  39.573254,  
  -104.87392  
],  
[  
  39.570576,  
  -104.873226  
],  
[  
  39.569907,  
  -104.873073  
],  
[  
  39.568179,  
  -104.872775  
],  
[  
  39.566606,  
  -104.872481  
],  
[  
  39.565383,  
  -104.872302  
],  
[  
  39.564188,  
  -104.872097  
],  
[  
  39.562196,  
  -104.871734  
],  
[  
  39.560067,  
  -104.871586  
],  
[  
  39.559809,  
  -104.871606
```

```
],  
[  
  39.559591,  
  -104.871597  
],  
[  
  39.559362,  
  -104.871604  
],  
[  
  39.559009,  
  -104.87164  
],  
[  
  39.558491,  
  -104.871765  
],  
[  
  39.558212,  
  -104.87186  
],  
[  
  39.558075,  
  -104.871925  
],  
[  
  39.557766,  
  -104.872114  
],  
[  
  39.557493,  
  -104.87232  
],  
[  
  39.557351,  
  -104.872447  
],  
[  
  39.557148,
```

```
-104.872694  
],  
[  
  39.556987,  
  -104.872934  
],  
[  
  39.55689,  
  -104.873113  
],  
[  
  39.556773,  
  -104.873361  
],  
[  
  39.556641,  
  -104.873743  
],  
[  
  39.556546,  
  -104.874105  
],  
[  
  39.556493,  
  -104.874357  
],  
[  
  39.556427,  
  -104.874773  
],  
[  
  39.556388,  
  -104.875193  
],  
[  
  39.556375,  
  -104.875475  
],  
[
```

```
39.556375,  
-104.875735  
],  
[  
39.556389,  
-104.876269  
],  
[  
39.556433,  
-104.876641  
],  
[  
39.556541,  
-104.877256  
],  
[  
39.556603,  
-104.877475  
],  
[  
39.55666,  
-104.877601  
],  
[  
39.556717,  
-104.877773  
],  
[  
39.556921,  
-104.878307  
],  
[  
39.557012,  
-104.87852  
],  
[  
39.557092,  
-104.8787  
],
```

```
[  
  39.557417,  
  -104.879366  
],  
[  
  39.557648,  
  -104.879778  
],  
[  
  39.557996,  
  -104.880334  
],  
[  
  39.558402,  
  -104.880866  
],  
[  
  39.558755,  
  -104.881296  
],  
[  
  39.558791,  
  -104.881385  
],  
[  
  39.559571,  
  -104.882188  
],  
[  
  39.560781,  
  -104.883397  
],  
[  
  39.561265,  
  -104.88394  
],  
[  
  39.561682,  
  -104.884484
```

```
],  
[  
  39.562036,  
  -104.885004  
],  
[  
  39.562613,  
  -104.885934  
],  
[  
  39.562819,  
  -104.886307  
],  
[  
  39.563028,  
  -104.88674  
],  
[  
  39.563363,  
  -104.887512  
],  
[  
  39.563575,  
  -104.888074  
],  
[  
  39.563776,  
  -104.888692  
],  
[  
  39.564005,  
  -104.889491  
],  
[  
  39.564154,  
  -104.890151  
],  
[  
  39.564282,
```



```
-104.890815  
],  
[  
  39.564324,  
  -104.891094  
],  
[  
  39.564431,  
  -104.891952  
],  
[  
  39.564469,  
  -104.892404  
],  
[  
  39.56449,  
  -104.89277  
],  
[  
  39.564509,  
  -104.89336  
],  
[  
  39.564506,  
  -104.893868  
],  
[  
  39.564494,  
  -104.894277  
],  
[  
  39.56432,  
  -104.897284  
],  
[  
  39.564262,  
  -104.898156  
],  
[
```

```
39.564222,  
-104.898571  
],  
[  
39.563996,  
-104.902115  
],  
[  
39.563949,  
-104.903002  
],  
[  
39.563927,  
-104.903795  
],  
[  
39.563926,  
-104.905055  
],  
[  
39.563946,  
-104.905853  
],  
[  
39.563978,  
-104.906528  
],  
[  
39.564021,  
-104.90715  
],  
[  
39.564151,  
-104.908672  
],  
[  
39.564603,  
-104.913959  
],
```

```
[  
  39.564679,  
  -104.914921  
],  
[  
  39.564709,  
  -104.915566  
],  
[  
  39.564713,  
  -104.916272  
],  
[  
  39.564685,  
  -104.917053  
],  
[  
  39.56461,  
  -104.917896  
],  
[  
  39.56454,  
  -104.918463  
],  
[  
  39.564433,  
  -104.919116  
],  
[  
  39.564334,  
  -104.919645  
],  
[  
  39.564279,  
  -104.919892  
],  
[  
  39.564085,  
  -104.920656
```

```
],  
[  
  39.56358,  
  -104.922343  
],  
[  
  39.563047,  
  -104.924051  
],  
[  
  39.562581,  
  -104.9256  
],  
[  
  39.562513,  
  -104.925872  
],  
[  
  39.561969,  
  -104.927711  
],  
[  
  39.561207,  
  -104.930202  
],  
[  
  39.561089,  
  -104.930618  
],  
[  
  39.560923,  
  -104.931248  
],  
[  
  39.560854,  
  -104.931542  
],  
[  
  39.560673,
```

```
-104.932438  
],  
[  
  39.560553,  
  -104.93317  
],  
[  
  39.560457,  
  -104.933966  
],  
[  
  39.560388,  
  -104.934818  
],  
[  
  39.560352,  
  -104.935834  
],  
[  
  39.56035,  
  -104.936385  
],  
[  
  39.560381,  
  -104.937217  
],  
[  
  39.560436,  
  -104.937988  
],  
[  
  39.560476,  
  -104.938377  
],  
[  
  39.560573,  
  -104.939113  
],  
[
```

```
39.560629,  
-104.939478  
],  
[  
39.560784,  
-104.94029  
],  
[  
39.560874,  
-104.940692  
],  
[  
39.560971,  
-104.941079  
],  
[  
39.561165,  
-104.941776  
],  
[  
39.561285,  
-104.94217  
],  
[  
39.561581,  
-104.943046  
],  
[  
39.561672,  
-104.943287  
],  
[  
39.562537,  
-104.945391  
],  
[  
39.562829,  
-104.946147  
],
```

```
[  
  39.563038,  
  -104.946771  
],  
[  
  39.563202,  
  -104.947338  
],  
[  
  39.563356,  
  -104.947968  
],  
[  
  39.563461,  
  -104.948464  
],  
[  
  39.563615,  
  -104.949263  
],  
[  
  39.56367,  
  -104.949652  
],  
[  
  39.563707,  
  -104.949982  
],  
[  
  39.563758,  
  -104.950558  
],  
[  
  39.563778,  
  -104.95088  
],  
[  
  39.563799,  
  -104.951556
```

```
],  
[  
  39.563792,  
  -104.952358  
],  
[  
  39.563803,  
  -104.952987  
],  
[  
  39.563741,  
  -104.954355  
],  
[  
  39.56369,  
  -104.955212  
],  
[  
  39.563664,  
  -104.956096  
],  
[  
  39.563678,  
  -104.958664  
],  
[  
  39.563691,  
  -104.959444  
],  
[  
  39.563727,  
  -104.959664  
],  
[  
  39.563768,  
  -104.95985  
],  
[  
  39.563818,
```



```
-104.959997  
],  
[  
  39.563854,  
  -104.960062  
],  
[  
  39.563934,  
  -104.96017  
],  
[  
  39.564019,  
  -104.960242  
],  
[  
  39.564163,  
  -104.960329  
],  
[  
  39.564323,  
  -104.960405  
],  
[  
  39.566238,  
  -104.960393  
],  
[  
  39.566227,  
  -104.962317  
],  
[  
  39.566145,  
  -104.962442  
],  
[  
  39.566124,  
  -104.965901  
],  
[
```

```
39.566145,  
-104.962442  
],  
[  
39.566081,  
-104.962313  
],  
[  
39.566079,  
-104.960614  
],  
[  
39.564646,  
-104.960624  
],  
[  
39.563714,  
-104.960648  
],  
[  
39.563092,  
-104.960649  
],  
[  
39.562693,  
-104.960626  
],  
[  
39.562547,  
-104.960605  
],  
[  
39.562303,  
-104.960552  
],  
[  
39.562363,  
-104.960316  
],
```

```
[  
  39.562531,  
  -104.959752  
],  
[  
  39.562648,  
  -104.959311  
],  
[  
  39.562721,  
  -104.959004  
],  
[  
  39.56277,  
  -104.95876  
],  
[  
  39.562881,  
  -104.958111  
],  
[  
  39.563027,  
  -104.957185  
],  
[  
  39.563132,  
  -104.95658  
],  
[  
  39.563198,  
  -104.956139  
],  
[  
  39.563329,  
  -104.955096  
],  
[  
  39.563358,  
  -104.954537
```

```
],  
[  
  39.563415,  
  -104.953687  
],  
[  
  39.56348,  
  -104.952476  
],  
[  
  39.563493,  
  -104.95189  
],  
[  
  39.563482,  
  -104.95137  
],  
[  
  39.563441,  
  -104.950599  
],  
[  
  39.563415,  
  -104.950372  
],  
[  
  39.56337,  
  -104.949783  
],  
[  
  39.563281,  
  -104.949088  
],  
[  
  39.563229,  
  -104.948775  
],  
[  
  39.563155,
```

```
-104.948392  
],  
[  
  39.563074,  
  -104.94802  
],  
[  
  39.562991,  
  -104.947679  
],  
[  
  39.56283,  
  -104.947094  
],  
[  
  39.562589,  
  -104.946344  
],  
[  
  39.562358,  
  -104.945724  
],  
[  
  39.561409,  
  -104.943408  
],  
[  
  39.561237,  
  -104.942943  
],  
[  
  39.560991,  
  -104.942198  
],  
[  
  39.560874,  
  -104.941803  
],  
[
```

```
39.560769,  
-104.941411  
],  
[  
39.56062,  
-104.94081  
],  
[  
39.560442,  
-104.939966  
],  
[  
39.560307,  
-104.939155  
],  
[  
39.560207,  
-104.938368  
],  
[  
39.560174,  
-104.938034  
],  
[  
39.560146,  
-104.937741  
],  
[  
39.56011,  
-104.937188  
],  
[  
39.560096,  
-104.936867  
],  
[  
39.560085,  
-104.936258  
],
```

```
[  
  39.560091,  
  -104.935483  
],  
[  
  39.560131,  
  -104.934643  
],  
[  
  39.560177,  
  -104.934079  
],  
[  
  39.560209,  
  -104.933781  
],  
[  
  39.560332,  
  -104.932838  
],  
[  
  39.560465,  
  -104.932084  
],  
[  
  39.560623,  
  -104.931341  
],  
[  
  39.560714,  
  -104.930963  
],  
[  
  39.560915,  
  -104.930232  
],  
[  
  39.561826,  
  -104.927213
```

```
],  
[  
  39.562324,  
  -104.925465  
],  
[  
  39.562879,  
  -104.923723  
],  
[  
  39.563802,  
  -104.920649  
],  
[  
  39.563863,  
  -104.920476  
],  
[  
  39.564024,  
  -104.919792  
],  
[  
  39.564132,  
  -104.919281  
],  
[  
  39.564242,  
  -104.918658  
],  
[  
  39.564327,  
  -104.91806  
],  
[  
  39.564399,  
  -104.91729  
],  
[  
  39.564437,
```



```
-104.9166  
],  
[  
  39.564447,  
  -104.916185  
],  
[  
  39.564444,  
  -104.91569  
],  
[  
  39.564428,  
  -104.915223  
],  
[  
  39.564353,  
  -104.914241  
],  
[  
  39.564029,  
  -104.910582  
],  
[  
  39.563834,  
  -104.909136  
],  
[  
  39.563389,  
  -104.906296  
],  
[  
  39.563189,  
  -104.905275  
],  
[  
  39.563072,  
  -104.904893  
],  
[
```

```
39.563006,  
-104.90478  
],  
[  
39.562964,  
-104.904735  
],  
[  
39.562873,  
-104.904663  
],  
[  
39.562756,  
-104.904605  
],  
[  
39.562408,  
-104.904533  
],  
[  
39.561631,  
-104.904548  
],  
[  
39.560285,  
-104.904553  
],  
[  
39.560171,  
-104.904563  
],  
[  
39.559854,  
-104.904631  
],  
[  
39.559745,  
-104.904665  
],
```

```
[  
  39.559547,  
  -104.904754  
],  
[  
  39.559373,  
  -104.904847  
],  
[  
  39.559135,  
  -104.905009  
],  
[  
  39.558852,  
  -104.905255  
],  
[  
  39.558692,  
  -104.905431  
],  
[  
  39.558521,  
  -104.905642  
],  
[  
  39.558424,  
  -104.905779  
],  
[  
  39.558327,  
  -104.905939  
],  
[  
  39.558157,  
  -104.90628  
],  
[  
  39.557999,  
  -104.906675
```

```
],  
[  
  39.558008,  
  -104.90668  
],  
[  
  39.557999,  
  -104.906675  
],  
[  
  39.557884,  
  -104.906969  
],  
[  
  39.557744,  
  -104.90736  
],  
[  
  39.557656,  
  -104.907554  
],  
[  
  39.557542,  
  -104.907707  
],  
[  
  39.557364,  
  -104.908206  
],  
[  
  39.556841,  
  -104.909613  
],  
[  
  39.556775,  
  -104.909775  
],  
[  
  39.556599,
```

```
-104.910148  
],  
[  
  39.556507,  
  -104.910309  
],  
[  
  39.556335,  
  -104.910566  
],  
[  
  39.556131,  
  -104.910815  
],  
[  
  39.556025,  
  -104.91093  
],  
[  
  39.555808,  
  -104.91113  
],  
[  
  39.555643,  
  -104.911255  
],  
[  
  39.555408,  
  -104.911397  
],  
[  
  39.555291,  
  -104.911451  
],  
[  
  39.554893,  
  -104.911592  
],  
[
```

```
39.554714,  
-104.911635  
],  
[  
39.554516,  
-104.911661  
],  
[  
39.554092,  
-104.911635  
],  
[  
39.549661,  
-104.910552  
],  
[  
39.547653,  
-104.91009  
],  
[  
39.547037,  
-104.910024  
],  
[  
39.546673,  
-104.910021  
],  
[  
39.546355,  
-104.910056  
],  
[  
39.546005,  
-104.910124  
],  
[  
39.545731,  
-104.910192  
],
```

```
[  
  39.545164,  
  -104.910399  
],  
[  
  39.544582,  
  -104.910691  
],  
[  
  39.544404,  
  -104.910809  
],  
[  
  39.544072,  
  -104.91106  
],  
[  
  39.54384,  
  -104.911257  
],  
[  
  39.543478,  
  -104.911615  
],  
[  
  39.543161,  
  -104.911984  
],  
[  
  39.543133,  
  -104.91216  
],  
[  
  39.542561,  
  -104.912841  
],  
[  
  39.542407,  
  -104.912868
```

```
],  
[  
  39.541946,  
  -104.913396  
],  
[  
  39.5418,  
  -104.91332  
],  
[  
  39.540717,  
  -104.911761  
],  
[  
  39.540289,  
  -104.911181  
],  
[  
  39.53984,  
  -104.910544  
],  
[  
  39.537512,  
  -104.907748  
],  
[  
  39.537293,  
  -104.907462  
],  
[  
  39.537098,  
  -104.907168  
],  
[  
  39.537006,  
  -104.907002  
],  
[  
  39.536875,
```



```
-104.906719  
],  
[  
  39.536739,  
  -104.906338  
],  
[  
  39.536695,  
  -104.906178  
],  
[  
  39.536609,  
  -104.905769  
],  
[  
  39.536561,  
  -104.905388  
],  
[  
  39.536547,  
  -104.905172  
],  
[  
  39.536535,  
  -104.904341  
],  
[  
  39.536485,  
  -104.904277  
],  
[  
  39.53642,  
  -104.902196  
],  
[  
  39.536367,  
  -104.897092  
],  
[
```

```
39.536375,  
-104.896131  
],  
[  
39.536302,  
-104.889481  
],  
[  
39.536272,  
-104.887199  
],  
[  
39.536252,  
-104.885474  
],  
[  
39.536229,  
-104.884475  
],  
[  
39.536256,  
-104.883495  
],  
[  
39.536259,  
-104.882187  
],  
[  
39.535689,  
-104.882169  
],  
[  
39.535598,  
-104.88213  
],  
[  
39.53522,  
-104.882083  
],
```

```
[
  39.535598,
  -104.88213
],
[
  39.535717,
  -104.882087
],
[
  39.535802,
  -104.882071
],
[
  39.536081,
  -104.882064
],
[
  39.536361,
  -104.882087
],
[
  39.536367,
  -104.882937
],
[
  39.536362,
  -104.885478
],
[
  39.539045,
  -104.885465
],
[
  39.540816,
  -104.885467
],
[
  39.541804,
  -104.885444
```

```
],  
[  
  39.543165,  
  -104.885435  
],  
[  
  39.545251,  
  -104.885437  
],  
[  
  39.54564,  
  -104.885455  
],  
[  
  39.545953,  
  -104.885493  
],  
[  
  39.546593,  
  -104.885611  
],  
[  
  39.546818,  
  -104.885643  
],  
[  
  39.547203,  
  -104.885663  
],  
[  
  39.547707,  
  -104.885635  
],  
[  
  39.547932,  
  -104.885607  
],  
[  
  39.548653,
```

```
-104.885461  
],  
[  
  39.549013,  
  -104.885419  
],  
[  
  39.549375,  
  -104.885404  
],  
[  
  39.550782,  
  -104.885404  
],  
[  
  39.552198,  
  -104.885456  
],  
[  
  39.552554,  
  -104.885439  
],  
[  
  39.552831,  
  -104.885397  
],  
[  
  39.553095,  
  -104.885328  
],  
[  
  39.553331,  
  -104.885249  
],  
[  
  39.553584,  
  -104.885147  
],  
[
```

```
39.553749,  
-104.885063  
],  
[  
39.554017,  
-104.884907  
],  
[  
39.554505,  
-104.884541  
],  
[  
39.554582,  
-104.884499  
],  
[  
39.55466,  
-104.884493  
],  
[  
39.555304,  
-104.883922  
],  
[  
39.555932,  
-104.883332  
],  
[  
39.555677,  
-104.88287  
]  
]  
}  
}
```