

# Package: plover (via r-universe)

September 6, 2024

**Title** What the Package Does (One Line, Title Case)

**Version** 0.0.1

**Description** Point in polygon.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Language** es

**LazyData** true

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.0

**LinkingTo** cpp11

**SystemRequirements** C++11

**URL** <https://github.com/hypertidy/plover>

**BugReports** <https://github.com/hypertidy/plover/issues>

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Imports** vctrs, wk

**Repository** <https://hypertidy.r-universe.dev>

**RemoteUrl** <https://github.com/hypertidy/plover>

**RemoteRef** HEAD

**RemoteSha** b707d33fad16848d78918e09cdc032e2b3878755

## Contents

ic_over	2
inside_clipper	2
inside_clipper_loop_mat	3
inside_clipper_loop_x_y	4
inside_orourke	4

<b>Index</b>	<b>6</b>
--------------	----------

---

ic_over	<i>insidclipper 'over(pts, polygons)' point in polygon lookup</i>
---------	---

---

**Description**

point in polygon

**Usage**

```
ic_over(pts, pol0, xyeps = NULL)
```

**Arguments**

pts	matrix of points
pol0	list of polygon matrices (or sf, sfc_M/POLYGON)
xyeps	offset,precision (WIP)

**Value**

integer of polygon

**Examples**

```
ic_over(matrix(runif(10), ncol = 2), list(c(0, .5, .5, 0, 0)))
```

---

inside_clipper	<i>Inside with clipper lib</i>
----------------	--------------------------------

---

**Description**

Call the clipper lib ...

**Usage**

```
inside_clipper(pts, coords, extent = NULL)
```

**Arguments**

pts	matrix of points 2 columns x,y
coords	matrix of polygon ring 2 columns x,y
extent	optional extents (WIP)

**Details**

WIP

**Value**

integer vector of point in polygon status, see Details

**Examples**

```
inside_clipper(matrix(runif(10), ncol = 2), cbind(c(0, .5, .5, 0, 0), c(0, 0, .5, 0, 0)))
```

---

`inside_clipper_loop_mat`

*Loop inside clipper polygon as matrixes*

---

**Description**

Loop clipper

**Usage**

```
inside_clipper_loop_mat(pts, lcoords, xyeps = NULL)
```

**Arguments**

<code>pts</code>	matrix of points
<code>lcoords</code>	list of matrixes
<code>xyeps</code>	offset x, offset y, precision

**Value**

list

**Examples**

```
inside_clipper_loop_mat(matrix(runif(10), ncol = 2),  
  list(cbind(c(0, .5, .5, 0, 0), c(0, 0, .5, 0, 0))))
```

---

```
inside_clipper_loop_x_y
```

*Loop inside clipper polygon as vectors*

---

### Description

Loop cliper

### Usage

```
inside_clipper_loop_x_y(pts, coordsx, coordsy)
```

### Arguments

pts	matrix of points
coordsx	list of vectors polygon x
coordsy	list of vectors polygon y

### Value

list

### Examples

```
inside_clipper_loop_x_y(matrix(runif(10), ncol = 2), list(c(0, .5, .5, 0, 0)),
list(c(0, 0, .5, 0, 0)))
```

---

```
inside_orourke
```

*Inside with O'Rourke's InPoly*

---

### Description

Call the O'Rourke InPoly function with cpp11.

### Usage

```
inside_orourke(pts, coords)
```

### Arguments

pts	matrix of points 2 columns x,y
coords	matrix of polygon ring 2 columns x,y

**Details**

For each query point 'pts', returns one of the following relative to P 'coords':  
0 : is strictly interior to P  
1 : is strictly exterior to P  
3 : is a vertex of P  
2 : lies on the relative interior of an edge of P

**Value**

integer vector of point in polygon status, see Details

**NOTE**

this was called `inside_cpp11` in the `insidecpp11` package

**Examples**

```
inside_orourke(matrix(runif(10), ncol = 2), cbind(c(0, .5, .5, 0, 0), c(0, 0, .5, 0, 0)))
```

# Index

ic\_over, 2  
inside\_clipper, 2  
inside\_clipper\_loop\_mat, 3  
inside\_clipper\_loop\_x\_y, 4  
inside\_orourke, 4