Ruby Array Patterns Matching

Pattern matching is an experimental feature allowing deep matching of structured values: checking the structure, and binding the matched parts to local variables.

Array patterns match arrays but only the whole array:

```
2.7.0 :001 > case ['a', 'b', 1]
2.7.0 :002 >   in [String, String, Integer]
2.7.0 :003 >     'match'
2.7.0 :004 > else
2.7.0 :005 >   'no match'
2.7.0 :006 > end
=> "match"
```

You can use the rest specification

```
2.7.0 :007 > case ['a', 'b', 1]
2.7.0 :008 > in [String, *]
2.7.0 :009 >     'match'
2.7.0 :010 > else
2.7.0 :011 >   'no match'
2.7.0 :012 > end
=> "match"
```

You can bind the matched part to a local variable:

```
2.7.0 :013 > case ['a', 'b', 1]
2.7.0 :014 > in [String => x, *]
2.7.0 :015 >     "match #{x}"
2.7.0 :016 > else
2.7.0 :017 >   'no match'
2.7.0 :018 > end
=> "match a"
```

You can simplify this:

```
2.7.0 :061 > case ['a', 'b', 'c']
2.7.0 :062 > in x, String, String
2.7.0 :063 >     "match #{x}"
2.7.0 :064 > else
2.7.0 :065 >   'no match'
2.7.0 :066 > end
=> "match a"
```

Or:

```
2.7.0 :067 > case ['a', 'b', 'c']
2.7.0 :068 > in [x, *]
2.7.0 :069 >     "match #{x}"
2.7.0 :070 > else
2.7.0 :071 >   'no match'
2.7.0 :072 > end
=> "match a"
```

Or:

```
2.7.0 :073 > case ['a', 'b', 'c']
2.7.0 :074 > in x, *rest
2.7.0 :075 >     "match #{x} and the rest: #{rest}"
2.7.0 :076 > else
2.7.0 :077 >   'no match'
2.7.0 :078 > end
=> "match a and the rest: [\"b\", \"c\"]"
```