

Ruby - Bug #18247

weird results for `Array#slice` or `Array#[]` with argument of type `Enumerator::ArithmeticSequence`

10/08/2021 03:38 PM - lxxxvi (Mario Schüttel)

Status:	Closed	Backport: 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:		

Description

There are weird results for Array#slice or Array#[] with argument of type Enumerator::ArithmeticSequence.

Particularly most results with negative steps are weird. For example:

```
@array = [0, 1, 2, 3, 4, 5]

@array.slice((2..).step(-1))      # => [2, 1, 0]
@array.slice((2..).step(-2))      # => [2, 0]
@array.slice((2..).step(-3))      # => [2]
@array.slice((2..).step(-4))      # => [0]          # ??? I'd have expected [2]

@array.slice((-3..).step(-1))     # => [3, 2, 1, 0]
@array.slice((-3..).step(-2))     # => [3, 1]
@array.slice((-3..).step(-3))     # => [3, 0]
@array.slice((-3..).step(-4))     # => [3]
@array.slice((-3..).step(-5))     # => [0]          # ??? I'd have expected [3]
```

Have a look at these examples with "beginless" ranges and negative steps.

Particularly the ones exluding the end value (...) seem wrong:

```
@array = [0, 1, 2, 3, 4, 5]

## end with zero index
@array.slice(..0).step(-1)        # => [5, 4, 3, 2, 1, 0]
@array.slice(...0).step(-1)       # => [5, 4, 3, 2, 1, 0]

@array.slice(..0).step(-2)        # => [5, 3, 1]
@array.slice(...0).step(-2)       # => [5, 3, 1]

@array.slice(..0).step(-10)       # => [0]
@array.slice(...0).step(-10)      # => [0]

## end with positive index
@array.slice(..3).step(-1)        # => [5, 4, 3]
@array.slice(...3).step(-1)       # => [5, 4, 3]

@array.slice(..3).step(-2)        # => [5, 3]
@array.slice(...3).step(-2)       # => [5, 3]

@array.slice(..3).step(-10)       # => [3]
@array.slice(...3).step(-10)      # => [3]

## end with negative index
@array.slice(..-2).step(-1)       # => [5, 4]
@array.slice(...-2).step(-1)      # => [5, 4]

@array.slice(..-2).step(-2)       # => [5]
@array.slice(...-2).step(-2)      # => [5]

@array.slice(..-2).step(-10)      # => [4]
@array.slice(...-2).step(-10)     # => [4]
```

Finally, some examples where the range is "inverted".

Particularly, the results for the ones excluding the end value (...) look strange.

But all examples seem weird, but I might not understand the concept of arithmetic sequence in depth.

```
@array = [0, 1, 2, 3, 4, 5]

# start and end with positive index
@array.slice((3..1).step(-1))      # => [3, 2, 1]
@array.slice((3...1).step(-1))    # => [2, 1]

@array.slice((3..1).step(-2))      # => [3, 1]
@array.slice((3...1).step(-2))    # => [2]

@array.slice((3..1).step(-10))     # => [1]
@array.slice((3...1).step(-10))   # => [1]

# start with negative index, end with positive index
@array.slice((-2..1).step(-1))     # => [4, 3, 2, 1]
@array.slice((-2...1).step(-1))    # => [3, 2, 1]

@array.slice((-2..1).step(-2))     # => [4, 2]
@array.slice((-2...1).step(-2))    # => [3, 1]

@array.slice((-2..1).step(-10))    # => [1]
@array.slice((-2...1).step(-10))   # => [1]

# start with positive index, end with negative index
@array.slice((4..-4).step(-1))     # => [4, 3, 2]
@array.slice((4...-4).step(-1))    # => [3, 2]

@array.slice((4..-4).step(-2))     # => [4, 2]
@array.slice((4...-4).step(-2))    # => [3]

@array.slice((4..-4).step(-10))    # => [2]
@array.slice((4...-4).step(-10))   # => [2]

# start with negative index, end with negative index
@array.slice((-2..-4).step(-1))    # => [4, 3, 2]
@array.slice((-2...-4).step(-1))   # => [3, 2]

@array.slice((-2..-4).step(-2))    # => [4, 2]
@array.slice((-2...-4).step(-2))   # => [3]

@array.slice((-2..-4).step(-10))   # => [2]
@array.slice((-2...-4).step(-10))  # => [2]
```

Found while writing specs for this method in <https://github.com/ruby/spec/pull/857>

Associated revisions

Revision cfb9624460a295e4e1723301486d89058c228e07 - 08/11/2022 10:16 AM - jeremyevans (Jeremy Evans)

Fix Array#[] with ArithmeticSequence with negative steps (#5739)

- Fix Array#[] with ArithmeticSequence with negative steps

Previously, Array#[] when called with an ArithmeticSequence with a negative step did not handle all cases correctly, especially cases involving infinite ranges, inverted ranges, and/or exclusive ends.

Fixes [Bug #18247]

- Add Array#slice tests for ArithmeticSequence with negative step to test_array

Add tests of rb_arithmetic_sequence_beg_len_step C-API function.

- Fix ext/-test-/arith_seq/beg_len_step/depend

- Rename local variables
- Fix a variable name

Co-authored-by: Kenta Murata 3959+mrkn@users.noreply.github.com

Revision cfb9624460a295e4e1723301486d89058c228e07 - 08/11/2022 10:16 AM - jeremyevans (Jeremy Evans)

Fix Array#[] with ArithmeticSequence with negative steps (#5739)

- Fix Array#[] with ArithmeticSequence with negative steps

Previously, Array#[] when called with an ArithmeticSequence with a negative step did not handle all cases correctly, especially cases involving infinite ranges, inverted ranges, and/or exclusive ends.

Fixes [Bug #18247]

- Add Array#slice tests for ArithmeticSequence with negative step to test_array

Add tests of rb_arithmetic_sequence_beg_len_step C-API function.

- Fix ext/-test-/arith_seq/beg_len_step/depend
- Rename local variables
- Fix a variable name

Co-authored-by: Kenta Murata 3959+mrkn@users.noreply.github.com

Revision cfb96244 - 08/11/2022 10:16 AM - jeremyevans (Jeremy Evans)

Fix Array#[] with ArithmeticSequence with negative steps (#5739)

- Fix Array#[] with ArithmeticSequence with negative steps

Previously, Array#[] when called with an ArithmeticSequence with a negative step did not handle all cases correctly, especially cases involving infinite ranges, inverted ranges, and/or exclusive ends.

Fixes [Bug #18247]

- Add Array#slice tests for ArithmeticSequence with negative step to test_array

Add tests of rb_arithmetic_sequence_beg_len_step C-API function.

- Fix ext/-test-/arith_seq/beg_len_step/depend
- Rename local variables
- Fix a variable name

Co-authored-by: Kenta Murata 3959+mrkn@users.noreply.github.com

History

#1 - 10/08/2021 03:56 PM - Eregon (Benoit Daloze)

- Description updated

#2 - 03/30/2022 11:05 PM - jeremyevans0 (Jeremy Evans)

This is still an issue in the master branch. I've submitted a pull request to fix it: <https://github.com/ruby/ruby/pull/5739>

#3 - 08/11/2022 10:17 AM - jeremyevans (Jeremy Evans)

- Status changed from Open to Closed

Applied in changeset [git|cfb9624460a295e4e1723301486d89058c228e07](https://github.com/ruby/ruby/pull/5739).

Fix `Array#[]` with `ArithmeticSequence` with negative steps ([#5739](#))

- Fix `Array#[]` with `ArithmeticSequence` with negative steps

Previously, `Array#[]` when called with an `ArithmeticSequence` with a negative step did not handle all cases correctly, especially cases involving infinite ranges, inverted ranges, and/or exclusive ends.

Fixes [Bug [#18247](#)]

- Add `Array#slice` tests for `ArithmeticSequence` with negative step to `test_array`

Add tests of `rb_arithmetic_sequence_beg_len_step` C-API function.

- Fix `ext/-test-/arith_seq/beg_len_step/depend`
- Rename local variables
- Fix a variable name

Co-authored-by: Kenta Murata 3959+mrkn@users.noreply.github.com