Ruby master - Bug #18956
Kernel#sprintf - %c handles negative Integer argument in a confusing way
08/06/2022 11:41 AM - andrykonchin (Andrew Konchin)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>ruby -v: 3.0.3</td>
</tr>
</tbody>
</table>

| Backport:  | 2.7: UNKNOWN, 3.0: UNKNOWN, 3.1: UNKNOWN |

**Description**

Integer argument for %c means a character codepoint.

I've noticed two outcomes when argument is negative:

- exception
- broken/incorrect string

When exception is raised - its message a bit misleading and confusing:

```
sprintf("%c", -1000)
# => invalid character (ArgumentError)
```

```
sprintf("%c".encode('ascii'), -1)
# => 4294967295 out of char range (RangeError)
```

invalid character means there is a character, but actual argument is a codepoint. 4294967295 out of char range is about codepoint, but it mentions 4294967295 instead of actual argument -1.

```
sprintf("%c", -1)
# => "\xFF"
```

In this case no exception is risen but the string is not valid:

```
sprintf("%c", -1).codepoints
# => invalid byte sequence in UTF-8 (ArgumentError)
sprintf("%c", -1).valid_encoding?
# => false
```

**Associated revisions**

Revision d2483393 - 08/20/2022 01:06 AM - nobu (Nobuyoshi Nakada)
[Bug #18956] Negative codepoints are invalid characters

**History**

#1 - 08/12/2022 02:50 PM - andrykonchin (Andrew Konchin)
- Description updated

#2 - 08/18/2022 09:44 AM - mame (Yusuke Endoh)
@naruse (Yui NARUSE), @nobu (Nobuyoshi Nakada), and @matz (Yukihiro Matsumoto) agreed that a negative integer to %c should raise an explicit exception.

#3 - 08/20/2022 01:06 AM - nobu (Nobuyoshi Nakada)
- Status changed from Open to Closed

Applied in changeset git|d2483393cbcb4dcfa0000fa8166bb7fa7ed9f7b4.

[Bug #18956] Negative codepoints are invalid characters