Discussion

Eduard Lebedyuk · Nov 17, 2021

Code Golf - Encoder

We need to send some coordinates to a spaceship through a laser beam.
To do that we have to encode it, and beam it out into space.
Your mission is to implement the encoder with a compression standard.
As usual shortest solution wins.

Task

You will receive a string of comma-separated integers and you will return a new string of comma-separated integers and sequence descriptors.

Input

"0,2,4,5,5,5,5,5,3,4,5"

Output

"0-4/2,5*5,3-5"

Note

- Compression happens left to right
- A sequence of 2 or more identical numbers is shortened as number*count
    - example: "5,5,5" is compressed to "5*3"
- A sequence of 3 or more consecutive numbers is shortened as first-last. This is true for both ascending and descending order
    - example: "1,3,4,5" is compressed to "1,3-5"
- A sequence of 3 or more numbers with the same interval is shortened as first-last/interval.
    - example: "0,2,4,6" is compressed to "0-6/2"
- Use this code to check the result length
- You also can use this test case here

Rules

1. The signature of the contest entry MUST be:

```
Class dc.golf.Encoder {

ClassMethod Compress(a)
{
}

}
```

2. It is forbidden to modify class/signature, including but not limited to:

- Adding inheritance
- Setting default argument values
- Adding class elements (Parameters, Methods, Includes, etc).

3. It is forbidden to refer to non-system code from your entry. For example, this is not a valid entry:

```
ClassMethod Compress(a)
{
  q ##class(myPackage.myClass).test(a)
}
```

4. The use of $ZWPACK and $ZWBPACK is also discouraged.

#Code Golf #Contest #InterSystems IRIS