Number sequence

Submission deadline:	**2021-11-07 23:59:59**
Late submission with malus:	**2022-01-02 23:59:59** (Late submission malus: 100.0000 %)
Evaluation:	**3.7500**
Max. assessment:	**5.0000** (Without bonus points)
Submissions:	5 / 20 Free retries + 10 Penalized retries (-10 % penalty each retry)
Advices:	4 / 2 Advices for free + 2 Advices with a penalty (-10 % penalty each advice)

The task is to develop a program that analyzes a sequence of integers.

Assume a sequence of non-negative integers (0, 1, 2, ...). The sequence starts with number zero, the numbers a re in an ascending order, and the numbers are adjacent to each other (no extra spaces or leading zeros). The beginning of the sequence is:

```
01234567891011121314151617181920...
```

The notation is not very readable for humans. To simplify further understanding, we colored the numbers like:

```
01234567891011121314151617181920...
```

The task of the program is to find the number and the particular digit in the given position in the sequence. For example:

- there is digit 0 of the number 0 in position 0,
- there is digit 2 of number 12 in position 15,
- there is digit 1 of number 710 in position 2021, ...

To make the program universal, we consider the sequences where the numbers are in various bases. For example, if the base i s 2, the (colored) sequence may look like:

```
01101110010111011110001001...
```

For example, there is digit 0 of number 110 in position 14 for the binary sequence.

Your program will be given a list of problems to solve in its standard input. Each problem consists of two decimal integers: the first number determines the position in the sequence, the second number is the base. The program computes the digit/number for each such problem and displays the result; the format is shown below. The program terminates when there are no further problems in the input, i.e., when the EOF is reached.

The program must validate input data. If the input is invalid, the program must detect it, it shall output an error message (see below), and terminate. If displayed, the error message must be sent to the standard output (do not send it to the error output) and the error message must be terminated by a newline (\n). The input is considered invalid, if:

- a position in the sequence is invalid (not an integer, negative position),
- the base is invalid (not an integer, outside of closed interval [2 ; 36]).

Sample program runs:

```
Position and radix:
0 10
0
^
3 10
3
^
9 10
9
^
10 10
10
^
11 10
10
 ^
12 10
11
^
13 10
11
 ^
189 10
99
 ^
190 10
100
```

```
^
191 10
100
 ^
192 10
100
  ^
193 10
101
^
```

Position and radix:

```
42 10
26
^
666 10
258
  ^
1889 10
666
 ^
12345 10
3363
   ^
1000000 10
185185
^
1234321 10
224238
   ^
```

Position and radix:

```
abc 10
Invalid input.
```

Position and radix:

```
12 37
Invalid input.
```

Advice:

- The sample runs above list both the output of your program (boldface font) and user input (regular font). The bold/regular formatting is included here, in the problem statement page, to increase readability of the listing. Your program must output the text without any additional markup.
- Do not forget the newline (\n) after the last output line.
- The program is tested with several tests where the program is tested for additional functions. A basic solution does not have to be extremely fast, moreover, the basic solution may ignore all bases except base 10 (for example, it may report an error if base is different from 10). Such a solution passes all mandatory tests and some optional tests. The result will be at most 75 %.
- An improved solution must handle the base parameter correctly. However, the solution does not have to be extremely fast (naive algorithm is still fine). Your program will be awarded nominal 100 % points if it passes the "various base" test (and all previous tests).
- Finally, there is a bonus speed test. There are big numbers in the input (sequence position is very high). The program does not have enough time to iterate over all intermediate numbers in the sequence. Instead, your program must use some smart algorithm that skips over some/all the intermediate numbers. If passed, the program is awarded more than the nominal 100 % points.
- The basic solution may use `int` data type to represent the values. However, the bonus test inputs big values that exceed the range of `int`. You need to use `long` or `long long` to pass the bonus test.
- Use lowercase letters to display digits in bases over 10. For example, `ff` is a valid result for 255 in base 16.
- Do not generate/store the sequence `01234567891011...` into a big string/array. It would be laborious, it would require a lot of memory, it would be slow, and it would require nontrivial programming. But most importantly, it is not needed. It may be a good idea to generate the intermediate numbers (e.g., in a for-loop), however, do not store them. In fact, you even do not need to convert the individual intermediate numbers into string.

Sample data: **Download**

Reference

- **Evaluator: computer**
  - Program compiled
  - Test 'Základní test s daty podle ukázky': success
    - result: 100.00 %, required: 100.00 %
    - Max. run time: 0.007 s (limit: 1.500 s)
    - Total run time: 0.026 s
    - Mandatory test success, evaluation: 100.00 %
  - Test 'Test mezních hodnot': success
    - result: 100.00 %, required: 50.00 %

        - Max. run time: 0.007 s (limit: 3.000 s)
        - Total run time: 0.033 s
        - Optional test success, evaluation: 100.00 %
    - Test 'Kontrola ošetření nesprávných vstupů': success
        - result: 100.00 %, required: 50.00 %
        - Max. run time: 0.006 s (limit: 0.200 s)
        - Total run time: 0.049 s
        - Optional test success, evaluation: 100.00 %
    - Test 'Test náhodnými daty': success
        - result: 100.00 %, required: 50.00 %
        - Max. run time: 0.007 s (limit: 1.000 s)
        - Total run time: 0.127 s
        - Optional test success, evaluation: 100.00 %
    - Test 'Test náodnými daty, různé číselné soustavy': success
        - result: 100.00 %, required: 75.00 %
        - Max. run time: 0.007 s (limit: 1.000 s)
        - Total run time: 0.129 s
        - Optional test success, evaluation: 100.00 %
    - Test 'Test rychlosti': success
        - result: 100.00 %, required: 100.00 %
        - Max. run time: 0.007 s (limit: 0.100 s)
        - Total run time: 0.128 s
        - Bonus test - success, evaluation: 125.00 %
    - Overall ratio: 125.00 % (= 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.25)
- Total percent: 125.00 %
- Early submission bonus: 0.50
- Total points: 1.25 * (5.00 + 0.50) = 6.88

SW metrics:

	Total	Average	Maximum	Function name
Functions:	**4**	--	--	--
Lines of code:	**78**	**19.50 ± 5.02**	**24**	`findDigit`
Cyclomatic complexity:	**17**	**4.25 ± 1.64**	**7**	`main`

5	**2021-11-02 17:17:47**	**Download**
Submission status:	Evaluated	
Evaluation:	3.7500	

- **Evaluator: computer**
    - Program compiled
    - Test 'Basic test with example data': success
        - result: 100.00 %, required: 100.00 %
        - Max. run time: 0.017 s (limit: 1.500 s)
        - Total run time: 0.034 s
        - Mandatory test success, evaluation: 100.00 %
    - Test 'Borderline test': success
        - result: 100.00 %, required: 50.00 %
        - Max. run time: 0.032 s (limit: 3.000 s)
        - Total run time: 0.054 s
        - Optional test success, evaluation: 100.00 %
    - Test 'Invalid input test': success
        - result: 100.00 %, required: 50.00 %
        - Max. run time: 0.006 s (limit: 0.200 s)
        - Total run time: 0.039 s
        - Optional test success, evaluation: 100.00 %
    - Test 'Random test': success
        - result: 100.00 %, required: 50.00 %
        - Max. run time: 0.008 s (limit: 1.000 s)
        - Total run time: 0.142 s
        - Optional test success, evaluation: 100.00 %
    - Test 'Random test, various base': failed
        - result: 10.00 %, required: 75.00 %
        - Max. run time: 0.006 s (limit: 1.000 s)
        - Total run time: 0.094 s
        - Optional test failed, evaluation: 75.00 %
        - ☑ Failed (invalid output)
            - ☑ Input data **[58 B / 58 B]**

```
604 2
748 2
542 2
568 2
634 2
403 2
17 2
827 2
```

```
117 2
21 2
```

Output data **[82 B / 82 B]**

```
Position and radix:
238
^
286
^
217
 ^
226
^
248
^
171
^
13
 ^
312
 ^
63
 ^
15
 ^
```

Reference **[140 B / 140 B]**

```
Position and radix:
1101000
  ^
1111100
      ^
1011111
   ^
1100011
 ^
1101100
    ^
1001011
    ^
111
  ^
10000111
 ^
11101
  ^
1000
   ^
```

- Failed (invalid output)

  Input data **[58 B / 58 B]**

```
895 8
456 8
375 8
521 8
139 8
664 8
409 8
258 8
1 8
225 8
```

  Output data **[87 B / 87 B]**

```
Position and radix:
335
^
188
  ^
161
  ^
210
 ^
74
 ^
258
```

```
^
173
^
122
  ^
1
^
111
  ^
```

Reference **[84 B / 84 B]**

```
Position and radix:
502
 ^
260
^
225
^
305
  ^
106
 ^
365
 ^
240
 ^
156
^
1
^
143
^
```

- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)
- Failed (invalid output)

- Test 'Speed test': Abnormal program termination (Time limit exceeded)
  - Program was killed after: 0.103 s (limit: 0.100 s)
  - Bonus test - failed, evaluation: No bonus awarded
  - Time limit exceeded
- Overall ratio: 75.00 % (= 1.00 * 1.00 * 1.00 * 1.00 * 0.75)

- Advices used: 2
- Penalty due to advices: None (2 <= 2 limit)
- Total percent: 75.00 %
- Total points: 0.75 * 5.00 = 3.75

SW metrics:

	Total	Average	Maximum	Function name
Functions:	**1**	--	--	--
Lines of code:	**72**	**72.00 ± 0.00**	**72**	`main`
Cyclomatic complexity:	**14**	**14.00 ± 0.00**	**14**	`main`

4	**2021-11-02 15:50:58**	**Download**
Submission status:	Evaluated	
Evaluation:	3.3750	

- **Evaluator: computer**
  - Compile in 'pedantic' mode failed (10 % penalty).
  - Test 'Basic test with example data': success
    - result: 100.00 %, required: 100.00 %
    - Max. run time: 0.015 s (limit: 1.500 s)
    - Total run time: 0.031 s
    - Mandatory test success, evaluation: 100.00 %

- Test 'Borderline test': success
  - result: 100.00 %, required: 50.00 %
  - Max. run time: 0.032 s (limit: 3.000 s)
  - Total run time: 0.056 s
  - Optional test success, evaluation: 100.00 %
- Test 'Invalid input test': success
  - result: 100.00 %, required: 50.00 %
  - Max. run time: 0.005 s (limit: 0.200 s)
  - Total run time: 0.038 s
  - Optional test success, evaluation: 100.00 %
- Test 'Random test': success
  - result: 100.00 %, required: 50.00 %
  - Max. run time: 0.008 s (limit: 1.000 s)
  - Total run time: 0.121 s
  - Optional test success, evaluation: 100.00 %
- Test 'Random test, various base': failed
  - result: 10.00 %, required: 75.00 %
  - Max. run time: 0.007 s (limit: 1.000 s)
  - Total run time: 0.097 s
  - Optional test failed, evaluation: 75.00 %
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - ☑ Failed (invalid output)
    - ☑ Input data **[80 B / 80 B]**

```
26601 2
92438 2
69059 2
41028 2
83481 2
86751 2
29771 2
45710 2
39737 2
41322 2
```

    - ☑ Output data **[35 B / 35 B]**

```
Position and radix:
Invalid input.
```

    - ☑ Reference **[227 B / 227 B]**

```
Position and radix:
100111111101
           ^

1111000111100
        ^

1011100110110
   ^

111010110000
  ^

1101110001011
        ^

1110010000111
  ^

101100000110
 ^

1000000110010
  ^

111001000100
        ^

111011001000
        ^
```

  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)
  - Failed (invalid output)

        - Failed (invalid output)
    - Test 'Speed test': Abnormal program termination (Time limit exceeded)
        - Program was killed after: 0.102 s (limit: 0.100 s)
        - Bonus test - failed, evaluation: No bonus awarded
        - Time limit exceeded
    - Overall ratio: 67.50 % (= (1.00 * 1.00 * 1.00 * 1.00 * 0.75) * 0.9)
- Advices used: 1
- Penalty due to advices: None (1 <= 2 limit)
- Total percent: 67.50 %
- Total points: 0.68 * 5.00 = 3.38

SW metrics:

	Total	Average	Maximum	Function name
Functions:	**1**	--	--	--
Lines of code:	**72**	**72.00 ± 0.00**	**72**	`main`
Cyclomatic complexity:	**13**	**13.00 ± 0.00**	**13**	`main`

3	**2021-10-30 11:34:48**	**Download**
Submission status:	Evaluated	
Evaluation:	0.0000	

- **Evaluator: computer**
    - Program compiled
    - Test 'Basic test with example data': failed
        - result: 50.00 %, required: 100.00 %
        - Max. run time: 0.007 s (limit: 1.500 s)
        - Total run time: 0.024 s
        - Mandatory test failed, evaluation: 0.00 %
        - ☑ Failed (invalid output)

☑ Input data **[74 B / 74 B]**

```
0 10
3 10
9 10
10 10
11 10
12 10
13 10
189 10
190 10
191 10
192 10
193 10
```

☑ Output data **[24 B / 24 B]**

```
Position and radix:
0
^
```

☑ Reference **[87 B / 87 B]**

```
Position and radix:
0
^
3
^
9
^
10
^
10
 ^
11
^
11
 ^
99
 ^
100
^
100
 ^
100
  ^
```

```
101
^
```

        - Failed (invalid output)
    - Overall ratio: 0.00 %
- Total percent: 0.00 %
- Early submission bonus: 0.50
- Total points: 0.00 * (5.00 + 0.50) = 0.00

SW metrics:		Total	Average	Maximum	Function name
	Functions:	**1**	--	--	--
	Lines of code:	**51**	**51.00 ± 0.00**	**51**	`main`
	Cyclomatic complexity:	**10**	**10.00 ± 0.00**	**10**	`main`

2	**2021-10-30 11:05:05**	**Download**
Submission status:	Evaluated	
Evaluation:	0.0000	

- **Evaluator: computer**
    - Program compiled
    - Test 'Basic test with example data': failed
        - result: 50.00 %, required: 100.00 %
        - Max. run time: 0.007 s (limit: 1.500 s)
        - Total run time: 0.021 s
        - Mandatory test failed, evaluation: 0.00 %
        - Failed (invalid output)
        - Failed (invalid output)
    - Overall ratio: 0.00 %
- Total percent: 0.00 %
- Early submission bonus: 0.50
- Total points: 0.00 * (5.00 + 0.50) = 0.00

SW metrics:		Total	Average	Maximum	Function name
	Functions:	**1**	--	--	--
	Lines of code:	**51**	**51.00 ± 0.00**	**51**	`main`
	Cyclomatic complexity:	**11**	**11.00 ± 0.00**	**11**	`main`

1	**2021-10-29 22:04:27**	**Download**
Submission status:	Evaluated	
Evaluation:	0.0000	

- **Evaluator: computer**
    - Compile in 'pedantic' mode failed (10 % penalty).
    - Test 'Basic test with example data': Abnormal program termination (Time limit exceeded)
        - Program was killed after: 1.505 s (limit: 1.500 s)
        - Mandatory test failed, evaluation: 0.00 %
        - Time limit exceeded
    - Overall ratio: 0.00 %
- Total percent: 0.00 %
- Early submission bonus: 0.50
- Total points: 0.00 * (5.00 + 0.50) = 0.00

SW metrics:		Total	Average	Maximum	Function name
	Functions:	**1**	--	--	--
	Lines of code:	**52**	**52.00 ± 0.00**	**52**	`main`
	Cyclomatic complexity:	**11**	**11.00 ± 0.00**	**11**	`main`