Work on project. Stage 3/4: Convert to any base

273 users solved this problem. Latest completion was **about 20 hours ago**.

Project: Number Base Converter

Hard 2 hours

Description

1 / 1 Prerequisites

In project

BigInteger 1

Ideally, we want to convert numbers in different bases, not only from or to the decimal system. In this stage, we will add support for converting numbers from any given source base to any target base. As there are 26 Latin letters and 10 digits, the maximum base is 26 + 10 = 36. So, the target and source base will be between 2 and 36.

Also, it might be more convenient for our users if the program didn't ask for the base before each conversion and instead remembered the previously input base. This way, the users will have to do much less typing when they need to convert a bunch of numbers from base A to base B.

Join a study group for the project Number Base Converter

Discuss your current project with fellow learners and help each other.

To convert a number from the source base to the target base, you should first convert it to the decimal system and then convert the decimal number to the target base.

Note that from this stage on, numbers might be larger than you expect, so you should use `BigInteger` instead of `Int` or `Long` to avoid errors.

Objectives

Your program should have a two-level menu:

- On the first level, the user sees the following prompt: `Enter two numbers in format: {source base} {target base} (To quit type /exit)`. Then, they input two numbers separated by a single space: source base and target base. The user also has the option to use the `/exit` command to quit the program.
- On the second level, the user sees another prompt: `Enter number in base {user source base} to convert to base {user target base} (To go back type /back)`, and inputs the number in the source base. The program outputs the message `Conversion result:` followed by the number in the target base. Then, the program asks for the new number to convert from the previously provided source base to the target base. To change the bases, the user can choose the `BACK` command and return to the first level menu to make the necessary changes.

Example

The greater-than symbol followed by a space (`> `) represents the user input. Note that it's not part of the input.

```
Enter two numbers in format: {source base} {target base} (To quit type /exit) > 10 2
Enter number in base 10 to convert to base 2 (To go back type /back) > 11
Conversion result: 1011

Enter number in base 10 to convert to base 2 (To go back type /back) > 18
Conversion result: 10010

Enter number in base 10 to convert to base 2 (To go back type /back) > 127
Conversion result: 1111111

Enter number in base 10 to convert to base 2 (To go back type /back) > 189344562689000108753301247
Conversion result: 10011100100111110100010101001111100011111011010110101011010010011111101000101110111111111

Enter number in base 10 to convert to base 2 (To go back type /back) > /back

Enter two numbers in format: {source base} {target base} (To quit type /exit) > 36 10
Enter number in base 36 to convert to base 10 (To go back type /back) > abcde
Conversion result: 17325410

Enter number in base 36 to convert to base 10 (To go back type /back) > 13a0
Conversion result: 50904

Enter number in base 36 to convert to base 10 (To go back type /back) > az
Conversion result: 395

Enter number in base 36 to convert to base 10 (To go back type /back) > /back

Enter two numbers in format: {source base} {target base} (To quit type /exit) > /exit
```

See hint

Write a program

Code Editor IDE

Java

```
1
```

Run Continue Start again (reset) Solutions (78)

Comments (13) Hints (6) Useful links (2) Solutions (78) Show discussion