Programming Exercise 10
Base Number Conversion

C# Step by Step

Three number systems common in computing are binary (base 2), octal (base 8), and decimal (base 10). We commonly use the base 10 number system in everyday life. The legal digits are [0, 1, 2, 3, 4, 5, 6, 7, 8, 9], and the places are ones, tens, hundreds, thousands, etc. This is equivalent to 10^n, where n is the place, i.e. $10^0 = 1, 10^1 = 10, 10^2 = 100$, etc. Machine code is commonly represented in base 2. The legal digits are [0, 1], and the places are ones, twos, fours, eights, etc. This is equivalent to 10^n, where n is the place, i.e. $2^0 = 1, 2^1 = 2, 2^2 = 4$, etc. Base 8 number systems are commonly used for operation system utilities and configuration files, and base 8 is not generally familiar to the public. The legal digits are [0, 1, 2, 3, 4, 5, 6, 7], and the places are ones, 64ths, 512ths, 4096ths, etc. This is equivalent to 8^n, where n is the place, i.e. $8^0 = 1, 8^1 = 8, 10^2 = 64$, etc.

This exercise consists of writing functions to convert a number between bases. Write six functions as described below. **Do not use any built-in library or third party library! You are required to hand build all of these functions.** Note that I have placed all of my functions in a `Util` class; this is a result of my own disfunctionality — you do not have to do this.

1. binary → octal
2. binary → decimal
3. octal → binary
4. octal → decimal
5. decimal → binary
6. decimal → octal

Use the code below as a skeleton. I have written the function headers. Your task is to write correct implementations. Here are some equivalents: decimal 15 = octal 17 = binary 1111; decimal 42 = octal 52 = binary 101010; decimal 99 = octal 143 = binary 1100011.

This code is extremely easy to write. The only math operators you need are addition, multiplication, integer division, and modular division. The challenge is thinking *how to solve the problems*! Spend two days figuring out how to do the conversion. This takes absolutely no programming knowledge whatsoever. Then, spend an hour or two writing the code.

```
1  class Program
2  {
3      static void Main(string[] args)
4      {
5
6          Console.Write("Please enter the integer to convert: ");
7          string n1 = Console.ReadLine();
8          int number = int.Parse(n1);
9
10         Console.Write("Please enter the base to convert from [2 | 8 | 10]: ");
11         string n2 = Console.ReadLine();
12         int from = int.Parse(n2);
13
14         Console.WriteLine($"Number: {number}, base: {from}");
15
```

```
16          int result = 0;
17          if (from == 10)
18          {
19              result = Util.dec2bin(number);
20              Console.WriteLine($"binary conversion is {result}");
21              result = Util.dec2oct(number);
22              Console.WriteLine($"octal conversion is {result}");
23          }
24          else if (from == 2)
25          {
26              result = Util.bin2dec(number);
27              Console.WriteLine($"decimal conversion is {result}");
28              result = Util.bin2oct(number);
29              Console.WriteLine($"octal conversion is {result}");
30          }
31          else if (from == 8)
32          {
33              result = Util.oct2bin(number);
34              Console.WriteLine($"binary conversion is {result}");
35              result = Util.oct2dec(number);
36              Console.WriteLine($"decimal conversion is {result}");
37          }
38          else
39              Console.WriteLine("Error in base to convert from");
40      }
41  }
```