Wrting An Algorithm For Converting Bases

Yijun Chen

October 11, 2015

Different devices use different base to function.Computers use binary, others may use decimal system or systems in other bases. Numbers in different system will be represented in completely different ways.For example, in binary, the number"50" becomes 110010, which has four extra digits. When we encounter a number like "11011011000101001011100110111111001", an algorithm that can convert a number from decimal to other base or convert into decimal would be very handy.

Representing Python Code in Your Assignment

This is a very convenient algorithm converting numbers in different bases. The output will be shown as "ans" and the input can be any random number you want. Here we use 213 for our input and we are converting 213 into a base 4 system. In python, we use hashtags to comment on our code. Computers won't run the content after hashtags. We are converting this number from right to left(we calculate the last digit first). "x" represents the remainder of the input divider by the base number. Firstly, 213 modulo 4 to the power of one, and we get the remainder of 213 divided by four, which is four. "y" represents each of the converted digit. Therefore, the last digit of the out will be 1. The computer will repeat the while loop until the sum of each y has equaled the input.

```
ans =""
#this is static:c doesnt change
a=213
c=213
#this is base
b=4
#this is the exponent:e will change
e=0

while b**e<c:
    x=a%(b**(e+1))
    y=x/(b**(e))
    ans=str(y)+ans
    a=a-x
    e=e+1

print ans
```

```
The output is 3111
```