

## modulo encrypt problem

**Source:** [http://coding.derkeiler.com/Archive/C\\_CPP/comp.lang.c/2004-10/2002.html](http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2004-10/2002.html)

---

**From:** john blackburn (*john.blackburnNOSPAMPLS\_at\_lintonhealy.co.uk*)

**Date:** 10/19/04

Date: Tue, 19 Oct 2004 10:35:35 +0100

Hi,

I am trying to get an extremely simple character string encryption function to work using modulo 10 arithmetic.

```
void ccencode(UNSIGNED8* pattern) {  
  
    UNSIGNED8 key[] = CCKEY;  
    UNSIGNED8 digval = 0;  
    UNSIGNED8 count;  
  
    for (count=0; count < (strlen(pattern)-4); count++) {  
  
        digval = ((* (pattern+strlen(pattern)-count-1)-'0')+key[strlen(pattern)-count-1]+digval)%10;  
        *(pattern+4+count) = digval+'0';  
    }  
  
} /* ccencode */
```

The pattern to be encrypted is a string of ASCII numeric digits. This function starts with the last digit and adds it to the corresponding digit in the key (the key array of integers is #defined elsewhere) and then modulo divides it by 10 to always end up with a digit in the range 0 – 9. The digit is then added to '0' to make an ASCII character and written to the pattern. the digit then accumulates into the next and so on.

Two other characteristics are that the top 4 digits of the pattern are deliberately unencrypted and the remainder are not only encrypted, but also reversed in sequence.

My problem is that the first 6 digits are processed correctly but the remainder do not. The next 3 encrypted digits are 1 greater than they should be and the remaining 3 are then way out.

Surely this cannot be a rounding effect as % is an integer division giving an integer remainder. The algorithm works fine on a Borland C++ compiler but not on gcc. Note that I have to run gcc in ISO mode for other reasons.

comp.lang.c: modulo encrypt problem

Any ideas what is wrong ?

Thanks in advance

John Blackburn