

```
#include <stdio.h>
#include <ctype.h>

/*-----*/
/* Read the input */
/*-----*/
int read_od( int od[] )
{
    int ch, n ;

    n = 0 ;
    do {
        ch = getchar() ;
        if (isdigit(ch)) {
            od[n] = ch - '0' ;
            n++ ;
        }
    } while ( isdigit(ch) ) ;

    /* Consume to end of line. */
    while ( ( ch != '\n' ) && ( ch != EOF ) ) { ch = getchar() ; }
    return n ;
}

/*-----*/
/* Check if the array is all 9's. */
/*-----*/
int all_nines( int od[], int n )
{
    int flag, i ;

    flag = 1 ;
    i = 0 ;

    while( flag && ( i < n ) ) {
        flag = ( od[i] == 9 ) ;
        if (flag) i++ ;
    }
    return flag ;
}

/*-----*/
/* Increment the number represented in the array. */
/*-----*/
void increment_with_carry( int od[], int n )
{
    int k, carry ;

    k = n-1 ;
    carry = 0 ;
    od[k]++ ;
    do {
        od[k] += carry ;
        carry = 0 ;
        if (od[k] == 10) {
            od[k] = 0 ;
            carry = 1 ;
        }
        k-- ;
    } while ( carry && (k>=0)) ;
}
```

```
/*-----*/
/* Recursive solution to the Odometer Palindrome problem. */
/*-----*/
long long do_palindrome( int od[], int n )
{
    int a ;

    if ( ( n == 0 ) || ( n == 1 ) ) return 0 ; /* Base case. */

    /* Nothing to do here. */
    if ( od[0] == od[n-1] ) return 10 * do_palindrome( &od[1], n-2 ) ;

    if ( od[0] > od[n-1] ) { /* Easy case. */
        a = od[0] - od[n-1] ;
        return ( 10 * do_palindrome( &od[1], n-2 ) + a ) ;
    }
    else { /* od[0] must be less than od[n-1] */
        if ( all_nines( &od[1], n-2 ) ) {
            return 10 + od[0] - od[n-1] + 1 ;
        }
        else {
            a = 10 + od[0] - od[n-1] ;
            od[n-1] = od[0] ;
            increment_with_carry( &od[1], n-2 ) ;
            return 10 * do_palindrome( &od[1], n-2 ) + a ;
        }
    }
}

/*-----*/
/* M A I N ( ) --- Input loop. */
/*-----*/
int main()
{
    int od[10] ;
    int n ;
    long long m ;

    do {
        n = read_od( od ) ;
        if ( n > 1 ) {
            m = do_palindrome( od, n ) ;
            printf("%lld\n", m ) ;
        }
    } while ( n > 1 ) ;
}

===== Sample Session =====

sunset% cat in
100000
100001
000121
00456
0
sunset% gcc -o odo odometer.c
sunset% ./odo < in
1
0
979
44
```