

題目

將一正整數 n 的每一位數字向左移一位，並且將最左邊的數字移到個位，可以形成一個新的整數，例如 123 變成 231，4589 變成 5894，300 變成 003。現在輸入 n 及 k ，將 n 依上面之旋轉方式重覆做 k 次，則輸出之數字為何？（最前面的 0 不必列印，例如 030 印出 30 即可）

解題思惟

1. 考慮正整數 n ，例如 $n=4589$ ， n 的最左邊一位為千位，怎麼找出這一位呢？可以設 $d=1$ ，然後把 d 每次乘以 10（往左移一位），如果乘了之後會超過 n ，就不要再乘了，這時候的 n 就是最高位了。例如上述的例子， d 變成 1000 的時候，再乘會超過 n ，就不要再乘了，這是的 1000 就是 n 的最高位。以下程式碼可以找出 n 的最高位。

```
int d=1;
while (d*10<=n) d*=10;
```

2. 找到最高位之後，就可以知道 n 的最左位數字為 n/d 。除掉最左位的數為 $n\%d$ （為什麼？）這時候，上面的旋轉運算應該會得到一個新的數 $(n/d)+10*(n\%d)$ （為什麼？）
3. 因為旋轉的時候，還是保留相同位數在運算，所以 d 都不用變動。把第 2 個步驟做 k 次，這一題就解出來了。

程式碼

```
#include <stdio.h>
int main()
{
    int i, n, k, d=1;
    scanf("%d%d", &n, &k);
    while (d*10<=n) d*=10;
    for (i=0; i<k; i++) n = (n/d) + 10*(n%d);
    printf("%d", n);
    return 0;
}
```