Problem 6: Revised specification.

Write and test a program to meet this specification:

```
/*===========================================================================
 n <-- RLOG2(R,k)

Rational number logarithm, base 2.

Inputs
  R : a positive rational number,
  k : a positive BETA-digit.

Output
  n : an integer such that | L - n/2^k | < 1/2^k.

===========================================================================*/
#include "saclib.h"

Word RLOG2(R,k)
       Word R,k;
{
Word n;
   ///
}
```

Hint;

Note that the integer part, m, of $\log_2(R)$ can be computed by use of `ILOG2()`. Then $1 \leq R/2^m < 2$. Let $S = R/2^k$. It suffices to compute successive bits of $l = \log_2(S)$. Notice that the first bit of l will be 1 just in case $S^2 \geq 2$. By adjusting and repeated squaring, additional bits may be determined similarly.

Try your program for several values of k. What do your experiments suggest about the efficiency of the algorithm? Give a brief report and discussion.