Problem 4(a): Converting numbers to binary form

> a := convert(654321, binary);

\( a := 1001111110111110001 \)  

(1)

> b := convert(654321, base, 2);

\[ b := \{1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1\} \]

(2)

The first command returns the binary expansion of 654321. (Note that this is actually an integer whose decimal expansion has only 0's and 1's.; cf. Maple's help apge on "convert/binary".)

The second command gives the list of binary digits (bits) of the binary expansion of 654321 in reverse order.

Problem 4(b): A program which converts a given integer \( n > 0 \) into its binary form (list of bits).

Input: an integer \( n > 0 \). Output: the list of bits of \( n \) in reverse order (as for the convert(*, base, 2) command).

> bink := proc(n) local ls, q, r;
ls := [ ]; q := n;
while (q \neq 0) do;
  r := irem(q, 2); q := iquo(q, 2);
  ls := [op(ls), r]; od;
return(ls); end;

bink := proc(n)
  local ls, q, r;
  ls := [ ]; q := n;
  while q < 0 do
    r := irem(q, 2); q := iquo(q, 2); ls := [op(ls), r];
  end do;
  return ls
end proc

Testing this for \( n = 654321 \) yields:

> bink(654321);

\[ \{1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1\} \]

(4)

By inspection we see that this is the same answer as obtained by the command convert(654321, base, 2).

A better way to check this is by using MAPLE's evalb command:

> evalb(bink(654321) = b);

\text{true}  

(5)