

機能仕様

本プログラムは10進小数を2進小数に変換する。

入力

外部から次の2つの10進数字列を入力する

- ① 変換対象となる10進小数を表わす10進数字列

例 $0.d_1d_2\cdots d_n$ ここに、 d_i は0～9の数字

入力時に指定するのは $d_1d_2\cdots d_n$ の部分のみである。
0.の部分は指定しない。

- ② 変換後の2進小数の桁数を指定する10進数字列

例 l_1l_2 ここに、 l_1, l_2 は0～9の数字

出力

“10進小数： $d_1d_2\cdots d_n$ ”

“2進小数(l_1l_2 桁)： $B_1B_2\cdots B_m$ ” B_i は “0” or “1”

“表示指定の長さ超えの有無：{“有り” | “無し”}”

「表示指定の長さ超えの有無」は変換後の2進小数の桁数が②で指定した長さを越えているか否かを示す。

詳細仕様

次の番号順にプログラムを実行する。

(1) データの入力

- ① 変換対象の 10 進数文字列を入力する。

$d_1d_2\cdots d_n$

- ② 表示する 2 進小数の桁数を指定する 10 進数文字列を入力する。

(2) 計算リストの準備

計算リストは 10 進小数→2 進小数変換のために使用するリスト (配列)。

計算リストは、変換対象の 10 進数の各桁の 10 進数字 d_1, d_2, \dots, d_n を 2 進数値に変換したものを値として持つ D_1, D_2, \dots, D_n と、その左端に桁上がり要素 D を加えた $n+1$ 個の要素からなるリスト (配列) である。

D, D_1, D_2, \dots, D_n

D の値は 0 にしておく。

(3) 結果表示文字列を初期化

変換結果の 2 進小数を表示する文字列変数を

`result_str`

とする。ここに、変数の長さは (1) の②で指定したものである。

次の (4) の計算の結果得られた 2 進小数表示文字をこの変数に保存する。

(4) 変換計算を実行

計算には、(2) で用意した計算リストを使用する。

計算方法は筆算と同じである。

まず、右端の要素 D_n に 2 をかけ桁上がりは横に置き、残りを D_n とする。次に $D_{n-1} \times 2$ の桁上りを横に置き、「その残り+前の桁上がり」を D_{n-1} とに置く。この計算を繰り返し、 $D_1 \times 2$ の桁上りを横に置き、「その残り+前の桁上がり」を D_1 に置く。そしてこの最後の桁上りを D に置く。この D を文字に変換して `result_str` 内に左詰めに置く。

次に、計算リストの D を 0 にして、計算リストに対して上記の計算を実行し、そのときの D を文字に変換して `result_str` 内に左詰めに追加する。

何回か繰り返し計算し、計算リストの桁上がり要素 (D) を除くすべての要素 (D, D_1, D_2, \dots, D_n) が 0 になったとき計算を終了する。

もう 1 つの計算終了条件は、`result_str` が一杯になったときである。

(5) 結果の表示

入力した 10 進数文字列と変換結果の `result_str`、それに計算の終了条件となった表示指定の長さ超えの有無をプリントする。

(6) プログラムを終了する