アルゴリズム B 演習 No. 8　　2017 年 12 月 8 日配布

## 8. 四則演算と冪乗

- 2 進法で表された二つの自然数の加算を考える．$k$ ビット（2 進法で $k$ 桁）の自然数の加算は，次の操作を $k$ 回繰り返すことで行われる．
  - 足されるビット $x$，足すビット $y$，下の桁からの繰上ビット $z$ に対して，

    $$x + y + z = 2c + r$$

    によって，足した結果ビット $r$，上の桁への繰上ビット $c$ を定める．ここで，$x, y, z, r, c \in \{0, 1\}$ である．

  このビット演算を単位に求めた計算量を**ビット演算量** (bit complexity) という．したがって，高々 $k$ ビットの二つの自然数の加算のビット演算量は高々 $k$ である．
- 高々 $k$ ビットの二つの自然数の減算は，高々 $k$ のビット演算量でできる（ただし，ビット演算を適切に拡張するものとする）．
- $k$ ビットの自然数と $l$ ビットの自然数の乗算は，筆算と同様に行うと，高々 $kl$ のビット演算量でできる．ここで，桁をずらす操作の計算量は小さいので無視した．
- $k$ ビットの自然数を $l$ ビットの自然数で割って商と剰余を求める計算は，筆算と同様に行うと，高々 $kl$ のビット演算量でできる．
- $x$ を乗法の定義された代数系（正確には半群）の要素，$n$ を自然数として，$x^n$ の計算を考える．素朴な方法は，順に $x^2, x^3, \ldots, x^n$ と，$x$ による乗算を $n-1$ 回繰り返す方法である．乗算回数がより少ない方法を以下で述べる．$n$ の 2 進展開を $n = (n_k \ldots n_1 n_0)_2$ とする．ただし，$n_k = 1$ とする．

  $$x_0 = x, \quad x_i = {x_{i-1}}^2 = x^{2^i} \quad (i = 1, 2, \ldots, k)$$

  と定める．この計算は $k$ 回の乗算でできる．このとき，

  $$x^n = \prod_{\substack{n_i = 1 \\ 0 \le i \le k}} x_i$$

  によって $x^n$ が計算できる．

  $$\nu(n) = \#\{i \mid n_i = 1, 0 \le i \le k\}$$

  とおくと，$x^n$ の計算に必要な乗算の回数は，$k + \nu(n) - 1$ 回である．この方法を**繰り返し二乗法**という．（繰り返し二乗法には，乗算の順序がこれと異なるものもある．）

**注意.** 自然数の乗算アルゴリズムには，筆算よりも高速な方法が知られている．例えば，高速フーリエ変換 (FFT) を用いることで，高々 $k$ ビットの二つの自然数の積を $O(k \log k \log\log k)$ のビット演算量で計算することが出来る (Schönhage-Strassen, 1971)．また，理論的にこれより少し高速な乗算アルゴリズムも最近得られた (Fürer, 2007)．除算についても，ニュートン法を用いることで乗算と同程度の計算量で計算できる．

**注意.** ここまで自然数の四則演算について述べたが，整数についても符号を適切に考慮すれば同様に計算できる．整数の四則演算の計算量については，符号の操作等に必要な計算量は無視して前ページと同じビット演算量を用いることにする．

**問題**

解答に際して，その問題より前にある問題の結果を用いてもよい．
四則演算のビット演算量として，前ページで述べたものを使うものとする．
8-1, 8-2 は，2 進法のまま計算すること．

8-1. 2 進法で表された二つの整数 11001 と 10111 の積を筆算で求めよ．

8-2. 2 進法で表された整数 110110101 を 1011 で割った商と剰余を筆算で求めよ．

8-3. 自然数 $n$ を 2 進法で表したときの桁数は $\lfloor \log_2 n \rfloor + 1$ であることを示せ．

8-4.（a）素朴な方法で $x^{41}$ を計算したときの乗算の回数を求めよ．
（b）繰り返し二乗法で $x^{41}$ を計算したときの乗算の回数を求めよ．

8-5.（a）繰り返し二乗法で $x^{47}$ を計算したときの乗算の回数を求めよ．
（b）繰り返し二乗法よりも少ない乗算回数で $x^{47}$ を計算する方法を一つ与えよ．

8-6. $p(X) = a_n X^n + \cdots + a_1 X + a_0$ を $X$ の整数係数多項式とする．整数 $x$ に対して，$b_0, b_1, \ldots, b_n$ を以下のように定める．

$$b_n = a_n, \quad b_i = b_{i+1} x + a_i \ (i = n-1, \ldots, 1, 0).$$

（a）$b_0 = p(x)$ が成り立つことを示せ．
（b）この方法で $p(x)$ を計算するとき，必要な乗算と加算の回数をそれぞれ求めよ．
（この方法を**ホーナー法**という．）

8-7. 8-6 において，$a_0, a_1, \ldots, a_n$ と $x$ は高々 $n$ ビットであるとする．8-6 の方法で $p(x)$ を計算するときのビット演算量は $O(n^4)$ であることを示せ．