

10 最大公約数

有用な関数 $\text{Gcd}(M, N)$ (m と n の最大公約数) を定義します。

10.1 解説

m を n で割った商を q , 余りを r とすると,

$$m - qn = r$$

d を m と n の公約数とすると

$$m = dm', n = dn'$$

となる m', n' がある。ゆえに

$$r = m - qn = dm' - qdn' = d(m' - qn')$$

となり, d は r の約数でもある。すなわち

$$d \text{ が } m, n \text{ の公約数} \Rightarrow d \text{ は } n, r \text{ の公約数}$$

逆も成り立つ。ゆえに,

$$d \text{ が } m, n \text{ の最大公約数} \Leftrightarrow d \text{ が } n, r \text{ の最大公約数}$$

したがって, m, n の最大公約数 $\text{gcd}(m, n)$ は次の例のようにして求められる。

例 10.1 (1001 と 413 の最大公約数)

m	n	$r = m \bmod n$	計算
1001	413	175	$1001 \div 413 = 2 \cdots 175$
↙	↙		
413	175	63	$413 \div 175 = 2 \cdots 63$
↙	↙		
175	63	49	$175 \div 63 = 2 \cdots 49$
↙	↙		
63	49	14	$63 \div 49 = 1 \cdots 14$
↙	↙		
49	14	7	$49 \div 14 = 3 \cdots 7$
↙	↙		
14	7	0	$14 \div 7 = 2 \cdots 0$
↙	↙		
7	0		

ゆえに, $\text{gcd}(1001, 413) = 7$ 。

10.2 プログラム

```
1 program SaidaiKouyakusuu; // 学生証番号 氏名
2 {$APPTYPE CONSOLE}
3 uses SysUtils;
4
5 function Gcd(M,N : Integer) : Integer;
6     { mとnの最大公約数 }
7     var
8         R : Integer;
9
10    begin
11        WriteLn('M':11, 'N':11, 'R':11);
12        Write(M:11, N:11);
13        repeat
14            R := M mod N;
15            WriteLn(R:11);
16            M := N;
17            N := R;
18            Write(M:11, N:11);
19        until N = 0;
20        WriteLn;
21        Gcd := M;
22    end; {Gcd}
23
24 var
25     A,B,D,M : Integer;
26
27 begin {Main}
28     WriteLn(' a と b の最大公約数と最小公倍数');
29     repeat
30         WriteLn;
31         Write(' a b [同じ数を入れると終わり] ? ');
32         ReadLn(A,B);
33         D := Gcd(A,B);
34         WriteLn(' 最大公約数 = ', D);
35         M := (A div D)*B;
36         WriteLn(' 最小公倍数 = ', M);
37     until A = B;
38     ReadLn;
39 end.
```

実行

- (1) $a = 1001, b = 413$ としなさい。
- (2) $a = 1447, b = 3355$ としなさい。
- (3) $a = 123, b = 0$ としなさい。

注 10.1 (2) のように, $a < b$ の場合も正しく計算します。

注 10.2 (3) のように, $b = 0$ とすると, 14 行目で $n = 0$ で割り算することになり実行時エラーになります。

10.3 修正

```
repeat
  文 1;
  :
  文 n;
until 終了条件
```

は、“文 1;...文 n;” を実行してから、“終了条件” を調べて、繰り返すか終了するか判断します。ゆえに、必ず 1 回は“文 1;...文 n;” を実行します。

“文 1;...文 n;” を 1 回も実行しないことを可能にするには、実行する前に繰り返すか終了するか判断する必要があります。それをするのが、while ループです。

```
while 実行条件 do
  begin
    文 1;
    :
    文 n;
  end
```

注 10.3 “実行条件” が真 (true) のとき “文 1;...文 n;” を実行します。ゆえに repeat ループの “終了条件” の否定を書きます。

“文 1;...文 n;” を begin end でくくります。

プログラム (関数定義部分のみ) もう途中経過を書く必要はないので Write 文は除去しました。

```
function Gcd(M,N : Integer) : Integer;
  { mとnの最大公約数 }
var
  R : Integer;
begin
  while N > 0 do
    begin
      R := M mod N;
      M := N;
      N := R;
    end;
  Gcd := M;
end; {Gcd}
```

