I222 計算の理論
“Theory of Computation”
Report (3)

Teacher: Ryuhei UEHARA TA: Sachio TERAMOTO

Japan Advanced Institute of Science and Technology

November 1st, 2006 Office Hour

問題 1

正整数上の関数 c を次のように定義する:

$$c(n) = \begin{cases} \frac{n}{2} & n \text{ が偶数のとき} \\ 3n+1 & n \text{ が奇数のとき} \end{cases}$$

また整数 k に対し，正整数上の関数 c^k を次のように定義する:

$$c^k(n) = \begin{cases} c(n) & k = 1 \text{ のとき} \\ c(c^{k-1}(n)) & k > 1 \text{ のとき} \end{cases}$$

ここで集合 $\mathcal{C}$ を以下のように定義する.

$$\mathcal{C} = \{n \mid \text{ある正整数 } k \text{ に対して } c^k(n) = 1 \text{ が成立する }\}$$

なお，コラッツの予想とは $\mathcal{C} = \mathbb{N}$ であり，この予想は $n < 3 \times 2^{53}$ までは成立することが確かめられている．コラッツの予想が成立するかどうかはわからないが，集合 $\mathcal{C}$ は枚挙可能である．$\mathcal{C}$ を枚挙するプログラムを示せ.

問題 1 の解答例

解答例

まず，次のプログラム A_C を考える:

```
prog A_C(input x);
 begin
   if x is not ⟨n,k⟩ for some n, k then
     halt(1);
   else if c^k(n) = 1 then
     halt(n);
   halt(1);
 end.
```

このとき，明らかに

- A_C は $\mathcal{C}$ に属する要素以外は出力しない
- 任意の $n \in \mathcal{C}$ に対して，ある k が存在して，$c^k(n) = 1$ となるので，これらを $\langle n, k \rangle$ の形で A_C に与えると，n が出力される．

したがって，$RANGE(A_C) = \mathcal{C}$ となる．

問題 1 の解答例 (つづき)

ここで $\langle k, n\rangle$ 上の順序 $\mathcal{O}$ を以下のように定義する.

$$
\begin{array}{cccc}
\langle 1,1\rangle & & & \\
\langle 1,2\rangle & \langle 2,1\rangle & & \\
\langle 1,3\rangle & \langle 2,2\rangle & \langle 3,1\rangle & \\
\langle 1,4\rangle & \langle 2,3\rangle & \langle 3,2\rangle & \langle 4,1\rangle \\
 & \vdots & &
\end{array}
$$

この順序 $\mathcal{O}$ 上での, i 番目の要素は計算可能である.
(プログラムの概略)

$i' = 1$
for ($t = 2$ to ∞){
  for ($s = 1$ to t) {
    $i' = i' + 1$ $//\ \langle s, t-s\rangle$ **が i' 番目の要素**
    if $i' = i$ then output $\langle s, t-s\rangle$
  }
}

一方, 任意の Σ 上の文字列 x に対して, x の長さ優先辞書式順序での順番 $(\mathrm{index}(x))$ は, 一意的に決まっていて, これも計算可能である.

問題 1 の解答例 (つづき)

以上を踏まえて, $\mathcal{C}$ の要素を一つずつ枚挙するプログラムの概略を以下に示す.

1. $\mathcal{S} = \varnothing$;
2. for $(i = 1$ to $\infty)$
3. 　　順序 $\mathcal{O}$ 上での i 番目の要素 $\langle k, n \rangle$ を計算する;
4. 　　if $A_{C}(\langle k, n \rangle) = n$ かつ $n \notin \mathcal{S}$ then
5. 　　　n を出力して, 集合 $\mathcal{S}$ に n を追加する;

コメント

上記のプログラムは,

- 任意の $n \in \mathcal{C}$ に対して, $A_{C}(\langle k, n \rangle) = n$ となる k が存在する.
  $\therefore i'$ 番目の $\langle k, n \rangle$ が $\mathcal{O}$ 上で計算されたときに n は出力される.
- $\mathcal{C}$ の要素以外は出力しない.
- $\mathcal{C}$ のある特定の要素が 2 回出力されることはないので, $\mathcal{C}$ の要素を枚挙するプログラムである.

問題 1 の解答例 (別解)

次の関数を考える.

$$\chi^i(n) = \begin{cases} 0 & \text{if } n = 1 \\ \frac{n}{2} & \text{if } i = 1 \text{ and } n \text{ is even} \\ 3n+1 & \text{if } i = 1 \text{ and } n (\neq 1) \text{ is odd} \\ \chi(\chi^{i-1}(n)) & \text{if } i > 1. \end{cases}$$

このとき,

$$n \in \mathcal{C} \iff [n = 1] \vee \exists k \ [\chi^k(n) = 1].$$

さらに, $n \in \mathcal{C}$ のとき, $n > 1$ であれば $[\chi^k(n) = 1]$ を満たす k は一意に定まる.

問題 1 の解答例 (別解つづき)

したがって, 次のプログラム `Collatz` は, $\mathcal{C}$ を枚挙する.

```
prog Collatz(input x);
 begin
   report(1);
   x := ε;
   while true do
      if (x is ⟨n, k⟩ for some n, k) ∧ (χ^k(n) = 1) then
         report(n);
      x := 長さ優先辞書式順序で x の次の文字列
   end-while
 end.
```

問題 2

HALT $\leq_m$ EVEN **を仮定すると**, $\mathcal{RE} \subseteq \mathcal{REC}$ **となることを示せ**.

戸田 誠之助 先生

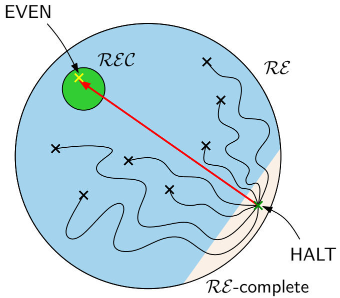
渡辺 治 先生

上原 隆平 先生

問題 2 の解答例

解答例

1. HALT は $\mathcal{RE}$ 完全集合である．したがって，任意の $\mathcal{RE}$ 集合 A に対して，帰納的還元 h_A が存在し，
$$x \in A \iff h_A(x) \in \text{HALT}.$$
2. HALT $\leq_m$ EVEN を仮定すると，HALT から EVEN への帰納的還元 h が存在し，
$$y \in \text{HALT} \iff h(y) \in \text{EVEN}.$$
3. EVEN $\in \mathcal{REC}$ であるので，任意の文字列 x に対して，$h(x) \in$ EVEN かどうかは認識可能である．



問題 2 の解答例

解答例

1. HALT は $\mathcal{RE}$ 完全集合である．したがって，任意の $\mathcal{RE}$ 集合 A に対して，帰納的還元 h_A が存在し，
$$x \in A \iff h_A(x) \in \text{HALT}.$$
2. $\text{HALT} \leq_m \text{EVEN}$ を仮定すると，HALT から EVEN への帰納的還元 h が存在し，
$$y \in \text{HALT} \iff h(y) \in \text{EVEN}.$$
3. $\text{EVEN} \in \mathcal{REC}$ であるので，任意の文字列 x に対して，$h(x) \in \text{EVEN}$ かどうかは認識可能である．

以上から，

$$\begin{aligned} x \in A &\iff h_A(x) \in \text{HALT} \\ &\iff h(h_A(x)) \in \text{EVEN} \end{aligned}$$

となり，しかも $h(h_A(x)) \in \text{EVEN}$ は認識可能である．よって $x \in A$ も認識可能であり，これは $A \in \mathcal{REC}$ を意味する．
任意の $\mathcal{RE}$ 集合 A に対して $A \in \mathcal{REC}$ が示されたので，$\mathcal{RE} \subseteq \mathcal{REC}$ を得る．

Problem 1

Let c be a function of a positive integer defined as follows:

$$c(n) = \begin{cases} \frac{n}{2} & n \text{ is even} \\ 3n+1 & n \text{ is odd} \end{cases}$$

The function c^k of a positive integer n for a positive integer k is defined as follows:

$$c^k(n) = \begin{cases} c(n) & \text{if } k = 1 \\ c(c^{k-1}(n)) & \text{if } k > 1 \end{cases}$$

Now we define a set $\mathcal{C}$ as follows:

$$\mathcal{C} = \{n \mid c^k(n) = 1 \text{ for some positive integer } k\}$$

Besides, Collatz conjectured that $\mathcal{C} = \mathbb{N}$, and it has been checked for all $n < 3 \times 2^{53}$ by brute force. We do not know whether the Collatz conjecture is true or not. However, the set $\mathcal{C}$ is enumerable. Show a program that enumerates the set $\mathcal{C}$.

Solution 1

Solution

First, let's consider the following program A_C :

```
prog A_C(input x);
 begin
   if x is not ⟨n, k⟩ for some n, k then
      halt(1);
   else if c^k(n) = 1 then
      halt(n);
   halt(1);
 end.
```

Then, obviously,

- A_C never reports any element not belonging in $\mathcal{C}$.
- For any $n \in \mathcal{C}$, there exists a k such that $c^k(n) = 1$.
  Hence, given an input as $\langle n, k \rangle$, the program A_C report n.

Therefore, we have $RANGE(A_C) = \mathcal{C}$.

Solution 1 (contd.)

We define our transiting order $\mathcal{O}$ on the tuple $\langle k, n \rangle$ as follows:

$$\begin{array}{llll} \langle 1,1\rangle & & & \\ \langle 1,2\rangle & \langle 2,1\rangle & & \\ \langle 1,3\rangle & \langle 2,2\rangle & \langle 3,1\rangle & \\ \langle 1,4\rangle & \langle 2,3\rangle & \langle 3,2\rangle & \langle 4,1\rangle \\ & \vdots & & \end{array}$$

The ith tuple on $\mathcal{O}$ is computable with the following program.
(Overview of Program)

```
i' = 1
for (t = 2 to ∞){
    for (s = 1 to t) {
        i' = i' + 1          // ⟨s, t − s⟩ is the i'th element
        if i' = i then output ⟨s, t − s⟩
    }
}
```

On the other hand, for any string x on Σ, the lexicographic order with length prefered of x is also computable, since it is identified uniquely.

Solution 1 (contd.)

Summing up, we can show the program that enumerates each element in $\mathcal{C}$ one by one.

1. $\mathcal{S} = \varnothing$;
2. for ($i = 1$ to ∞)
3. Compute the ith tuple $\langle k, n \rangle$ on our transiting order $\mathcal{O}$;
4. if $A_C(\langle k, n \rangle) = n$ and $n \notin \mathcal{S}$ then
5. Report n; $\mathcal{S} \leftarrow \mathcal{S} \cup \{n\}$;

Comment

The above program satisfies

- For an $n \in \mathcal{C}$, there exist k such that $A_C(\langle k, n \rangle) = n$
  $\therefore$ Hence, n is reported, once the i'th tuple $\langle k, n \rangle$ is computed on $\mathcal{O}$.
- never reports any element except for in $\mathcal{C}$.
- never duplicate to report a same element in $\mathcal{C}$.

Therefore, the program is for enumerating each element in $\mathcal{C}$.

Solution 1

Let's consider the following characteristic function instead of c^k :

$$\chi^i(n) = \begin{cases} 0 & \text{if } n = 1 \\ \frac{n}{2} & \text{if } i = 1 \text{ and } n \text{ is even} \\ 3n + 1 & \text{if } i = 1 \text{ and } n(\neq 1) \text{ is odd} \\ \chi(\chi^{i-1}(n)) & \text{if } i > 1. \end{cases}$$

Now we can redefine $\mathcal{C}$ by χ as follows:

$$n \in \mathcal{C} \iff [n = 1] \vee \exists k \; [\chi^k(n) = 1].$$

Moreover, if $n \in \mathcal{C}$ and $n > 1$, then we can uniquely identify a constant k such that $[\chi^k(n) = 1]$.

Solution 1 contd.

Therefore, the following program `Collatz` enumerates $\mathcal{C}$.

```
prog Collatz(input x);
begin
   report(1);
   x := ε;
   while true do
      if (x is ⟨n, k⟩ for some n, k) ∧ (χ^k(n) = 1) then
         report(n);
      x := the next word of x in the length prefered
            lexicographical order.
   end-while
end.
```

Problem 2

Suppose that HALT $\leq_m$ EVEN. Then show that we can obtain $\mathcal{RE} \subseteq \mathcal{REC}$.

Kurt Gödel

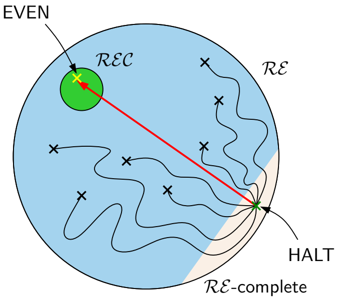
Alonzo Church

Alan Turing

Solution for problem 2

Solution

1. HALT is in $\mathcal{RE}$-complete. Hence, for any $A \in \mathcal{RE}$, there exists a reduction h_A such that
$$x \in A \iff h_A(x) \in \text{HALT}.$$
2. Supporting HALT $\leq_m$ EVEN, there also exists a reduction h from HALT to EVEN such that
$$y \in \text{HALT} \iff h(y) \in \text{EVEN}.$$
3. By EVEN $\in \mathcal{REC}$, for any $x \in \Sigma^*$, $[h(x) \in \text{EVEN}]$ is recognizable.



Solution for problem 2

Solution

1. HALT is in $\mathcal{RE}$-complete. Hence, for any $A \in \mathcal{RE}$, there exists a reduction h_A such that
$$x \in A \iff h_A(x) \in \text{HALT}.$$
2. Supporting HALT $\leq_m$ EVEN, there also exists a reduction h from HALT to EVEN such that
$$y \in \text{HALT} \iff h(y) \in \text{EVEN}.$$
3. By EVEN $\in \mathcal{REC}$, for any $x \in \Sigma^*$, $[h(x) \in \text{EVEN}]$ is recognizable.

- Finally,
$$\begin{aligned} x \in A \quad &\iff \quad h_A(x) \in \text{HALT} \\ &\iff \quad h(h_A(x)) \in \text{EVEN}, \end{aligned}$$
and $[h(h_A(x)) \in \text{EVEN}]$ is recognizable.
- Hence $[x \in A]$ is also recognizable, and this implies $A \in \mathcal{REC}$.
- For any $A \in \mathcal{RE}$, we have $A \in \mathcal{REC}$.
$\therefore$ we can obtain $\mathcal{RE} \subseteq \mathcal{REC}$.