

いろはちゃんコンテスト 2019 day3 え-えそらんぐ 解説

settyan117

2019/05/02

本文中の Brainfuck の仕様から、以下の機能を作ることができます。ただし、[A] で A 番地のメモリの値を指すものとします。また、メモリの値はすべて正である場合を考えます（この問題では正でない場合を考える必要はありません）。

メモリの値を 0 にする

A 番地にある値を 0 にするには、次のようにします。

```
while [A] が 0 でない
  [A] を 1 減らす
end while
```

以後この操作を [A] を 0 にする と表します。

値の加算 (繰り上がりなし)

繰り上がりを考慮せずに、A 番地にある値を B 番地に加算するには、次のようにします。

```
while [A] が 0 でない
  [A] を 1 減らす
  [B] を 1 増やす
end while
```

操作後、[A] は 0 になります。

以後この操作を $[B] \leftarrow [A]$ と表します。

値のコピー

A 番地にある値を B 番地にコピーするには、次のようにします。C 番地はバッファです。[C] ははじめ 0 であるものとします。

```
[B] , [C] ← [A] // 同時に加算
[A] ← [C]
```

以後この操作を [B]=[A] と表します。

条件分岐 1

A 番地の値が正の場合のみ何かを実行したいときには、次のようにします。

```
while [A] が 0 でない
    [A] を 0 にする
    // ここにしたい処理
end while
```

操作後、[A] は 0 になります。

以後この操作を if [A] が 0 でない ~end if と表します。

条件分岐 2

A 番地の値が 0 の場合のみ何かを実行したいときには、次のようにします。C 番地はフラグの役割です。[C] ははじめ 0 であるものとします。

```
[C] を 1 増やす
if [A] が 0 でない
    [C] を 1 減らす
end if
if [C] が 0 でない
    [C] を 1 減らす
    // ここにしたい処理
end if
```

操作後、[A] は 0 になります。

以後この操作を if [A] が 0 ~end if と表します。

1 桁繰り上がり 10 進加算器

繰り上がりを考慮して [A] 番地の値を [B] 番地に足したいときは、次のようにします。C 番地には桁上がり (0 または 1) が入ります。[A]+[B] の値を [D] にコピーし、[D] の値を 1 ずつ最大 9 回減らして、途中で 0 になる ($\Leftrightarrow [D] < 10$) かどうか判定をしています。[C] ははじめ 0 で、[A] と [B] は 0 以上 9 以下であるものとします。

```
[B] ← [A]
[D] = [B]
[E] を 9 増やす
while [E] が 0 でない
    [E] を 1 減らす
    [D'] = [D]
    if [D] が 0 でない
        [F] を 1 増やす
        [D'] を 1 減らす
    end if
    [D] = [D']
    if [D'] が 0
        [F] を 0 にする
        [E] を 0 にする
    end if
end while
if [F] が 0 でない
    [B] を 10 減らす
    [C] を 1 増やす
end if
```

操作後、[A] は 0 になります。

これらを組み合わせることで、Brainfuck で 10 進 3 桁+3 桁の加算を行うプログラムを作ることができます (入力では ASCII コードが与えられるため、適宜 '0' = 48 を足し引きするとよいです)。これでこの問題が解けました。