

COWJS

石元稜

何を作ったか

- **COW言語** → (トランスパイル) → JavaScript
- **メリット**
 - Cow言語熟練者がJavaScriptの開発チームに加わる時 (?)
 - ソースコードを難読化させたいとき
- **デメリット**
 - Cow言語熟練者以外は開発しづらい

COW言語とは

- 難解プログラミング言語の一つ
 - Brainfuck
 - Whitespace
 - Intercal
 - Malbolge
 - Cow
- チューリング完全
- ウシを念頭に置いて設計された言語

COW プログラミング言語は、ウシを念頭に置いて設計されました。

牛の語彙力はやや限られているため、牛が知っている単語だけを言語に組み込むのは当然のことに思われました。その結果、すべての指示は「moo」のバリエーションであり、彼らが本当に理解している唯一の単語です。

COWPROGRAMMING LANGUAGE

<https://bigzaphod.github.io/COW/>

COW言語とは

構成

- レジスタ1個
- 十分な長さのメモリ
- ポインタが1つ
- 命令12個
 - moO, mOo, MoO, MOo, oom, OOM, MOO, moo, mOO, OOO, MMM, Moo

言語仕様

code	命令	内容
0	moo	ポインタの指す値が 0 なら、対応する「Moo」にジャンプ
1	mOo	ポインタをデクリメント
2	moO	ポインタをインクリメント
3	mOO	現在のポインタの指す値と一致する code 番号の命令として実行
4	Moo	現在のポインタの指す値が 0 なら、 STDIN から単一の ASCII 文字を読み取り、現在のポインタが指す位置に格納。 現在のポインタが指す値が 0 でないなら、 現在のポインタが指す値に対応する ASCII 文字をSTDOUTに出力する。
5	MOo	ポインタの指す値をデクリメント

言語仕様

code	命令	内容
6	MoO	ポインタの指す値をインクリメント
7	MOO	ポインタの指す値が 0 なら、対応する「moo」にジャンプ
8	OOO	ポインタの指す値を 0 に代入
9	MMM	レジスタに値がない場合 現在のポインタの指す値をコピーする。 レジスタに値がある場合 その値を現在のポインタの位置 に貼り付け、レジスタをクリアする。
10	OOM	ポインタの指す値を整数として STDOUT に出力する。
11	oom	標準入力から整数を読み取り、それを現在のポインタの位置 に格納

COW言語でHello, World

[illegible]

COW言語でHello, World

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO

Moo

Moo

MoO MoO MoO

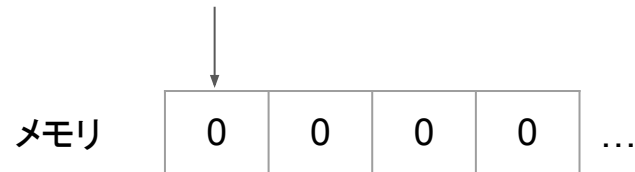
Moo

OOO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

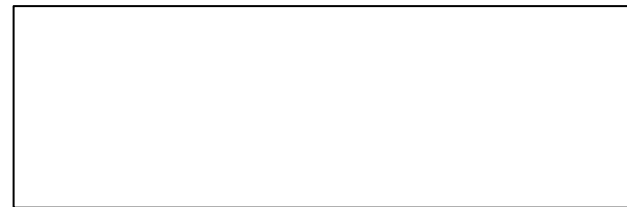
(省略)



レジスター



出力



MoO	ポインタの指す値をインクリメント
-----	------------------

COW言語でHello, World

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO

Moo

Moo

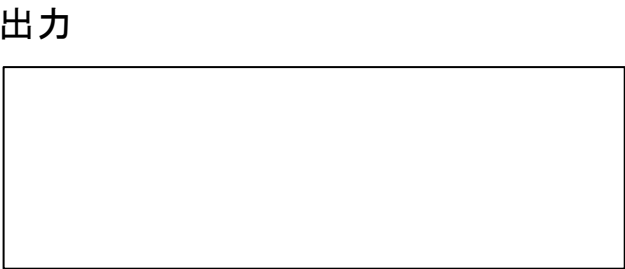
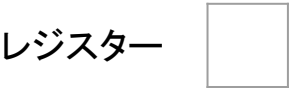
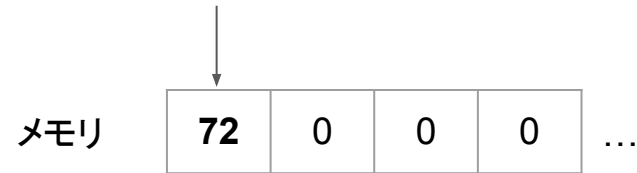
MoO MoO MoO

Moo

OOO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

(省略)



MoO	ポインタの指す値をインクリメント
-----	------------------

COW言語でHello, World

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO

Moo

Moo

MoO MoO MoO

Moo

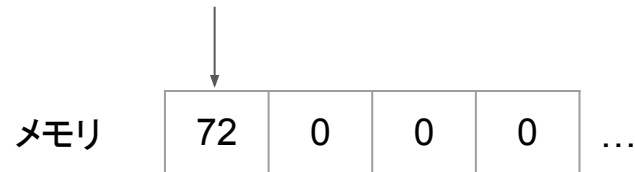
OOO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

(省略)



レジスター



出力

H

Moo	現在のメモリブロックが0 でない場合、現在のメモリブロックの値に対応するASCII 文字をSTDOUTに出力する。
-----	---

COW言語でHello, World

[illegible]

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO

Moo

Moo

MoO MoO MoO

Moo

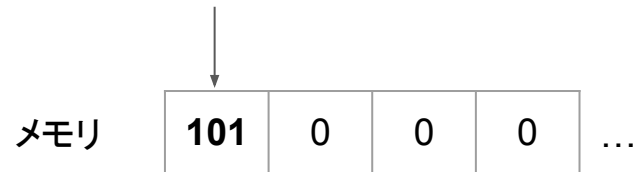
OOO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

(省略)



レジスター



出力

H

MoO	ポインタの指す値をインクリメント
-----	------------------

COW言語でHello, World

[illegible]

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO

Moo

Moo

MoO MoO MoO

Moo

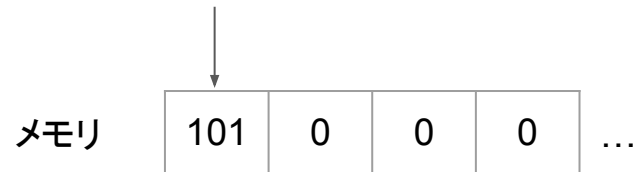
OOO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

(省略)



レジスター

出力

He

Moo	現在のメモリブロックが0 でない場合、現在のメモリブロックの値に対応するASCII 文字をSTDOUTに出力する。
-----	---

COW言語でHello, World

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO

Moo

Moo

MoO MoO MoO

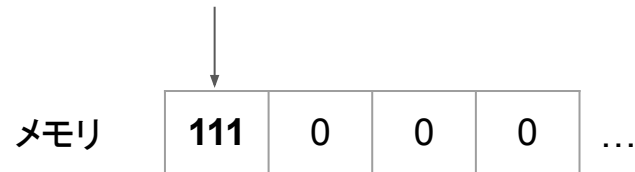
OOO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

(省略)



レジスター

出力

Hello

Moo	現在のメモリブロックが0 でない場合、現在のメモリブロックの値に対応するASCII 文字をSTDOUTに出力する。
-----	---

COW言語でHello, World

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO
MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO MoO

Moo

MoO MoO MoO MoO MoO MoO MoO

Moo

Moo

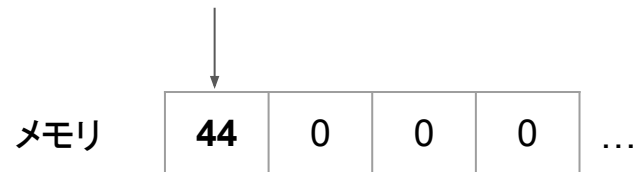
MoO MoO MoO

Moo

[illegible]

Moo

(省略)



レジスター



出力

Hello,

000	ポインタの指す値を0に代入
-----	---------------

COWからJavaScriptへ

[illegible]

変換

```
let pointer = 0;
let memory = new Array(40000).fill(0);
let register = null;
memory[pointer]++;
memory[pointer]++;
//(省略)
memory[pointer]++
if (memory[pointer]) {
    process.stdout.write(String.fromCharCode(
memory[pointer]));
} else {
    memory[pointer] = Number(require('fs').readFileSync(
'/dev/stdin', 'utf8').split('\n')[0]);
}
//(省略)
```

文法定義ファイル

```
start = body:line* { return emitProgram(body); }  
line = sp* instr:instruction* nl { return instr; }  
instruction = tkn:token sp* { return tkn; }  
token = moO / mOo / MoO / MOo / oom / OOM / Moo / MMM / OOO / MOO
```

```
moO = "moO" { return emitmoOToken(); }  
mOo = "mOo" { return emitmOoToken(); }  
MoO = "MoO" { return emitMoOToken(); }  
MOo = "MOo" { return emitMOoToken(); }  
oom = "oom" { return emitoomToken(); }  
OOM = "OOM" { return emitOOMToken(); }  
Moo = "Moo" { return emitMooToken(); }  
MMM = "MMM" { return emitMMMTToken(); }  
OOO = "OOO" { return emitOOOToken(); }  
MOO = "MOO" line:line* sp* instr:instruction* "moo"  
    { const content = line.concat(instr); return emitMOOToken(content); }  
sp = "\t" / " "  
nl = "\r" / "\n"
```


デモ

- fizzbuzz