

Runner

(開始時にヘッドの指していた文字を無限に書き続ける)

- これは "C言語による計算の理論(鹿島亮 著)" の4つの例題(suc, EQ_{ab}, BB3, Runner) の1つです.
- "C言語による計算の理論" の Turing machine は一度に「文字を1つ書くこと」と「左右に1つ進むこと(または動かないこと)」の2つの事ができます.
- 「2進法」も「aやbの個数」も使います. 終了する時は数字列の左端に移動する必要はありません.
- 遷移図は上記の本からお借りしました.
- TuringMachineの仕様は TuringMachine[rules,初期状態,step数] となります.
ruleは{q,s}→{q',s',dir} (q:現在の状態,s:ヘッドの下の文字,q':次の状態,s':ヘッドが書き込む文字,dir:ヘッドの進む方向.右,左,留の3通りで,+1,-1,0で指定)
初期状態は{{q0,pos},tape} (q0:最初の状態,pos:最初のヘッドの位置,tape:最初のテープの状態)となります.
- 例えば{1,a}→{2,b,+1} は「状態1でheadの文字がa」なら「ヘッドの下に文字bを書いて,ヘッドは右に1つ進み,内部状態は状態2へ移る」という事です.
- 同じ例は遷移図の方では「(1の書いてある丸)→(ab右)→(2の書いてある丸)」と表されます.
- tapeの初期状態は変数[tape]に入っています. 初期状態は変えることができますが,両端に空白がそれぞれ2個以上「最後まで残る」様にして下さい.
- 最後のManipulateではテープをクリックしてdragすると,画面の大きさが変わります.

■ 補助program

```
ClearAll["Global`*"]
```

(*ヘッドの位置を[]で,状態は添字で表示する*)

```
qbracket[x_List] := (*状態+ヘッドの位置. プログラムの検証にGood*)
```

```
ReplacePart[x[[2]], x[[1]][[2]] → Subscript["[" <> ToString[x[[2]][[x[[1]][[2]]]] <> "]", x[[1, 1]]]
```

(*状態FがTuringMachine[]の出力=outに最初に現れる位置.*)

```
posF[x_] := FirstPosition[x[[All, 1, 1]], F][[1]] - 1
```

```
turing[out_] :=
```

```
Manipulate[
```

```
Block[{now, tape, control, pos, state, contents, boxes},
```

```
now = out[[step]];
```

```
tape = now[[2]];
```

```
control = now[[1]];
```

```
pos = control[[2]];
```

```
state = control[[1]];
```

```
boxes = {Line[{{1, 0}, {1, 2}}], Table[Line[{{i, 0}, {i + 1, 0}, {i + 1, 2}, {i, 2}}], {i, 1, Length[tape]}};
```

```
contents = {Table[Text[Style[tape[[i]], Large], {i + .5, 1}], {i, 1, Length[tape]}],
```

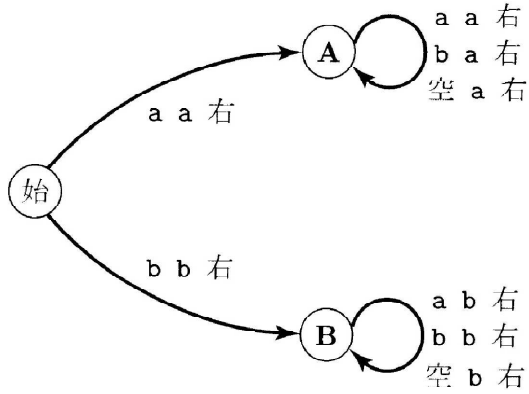
```
Green, Polygon[{{pos, -1}, {pos + 1, -1}, {pos + 1, -.5}, {pos + .5, 0}, {pos, -.5}}],
```

```
Black, Text[Style[state, Medium], {pos + 0.5, -0.7}]];
```

```
Graphics[{boxes, contents}], {step, 1, Length[out], 1}]
```

main program

```
Import["https://mixedmoss.com/mathematica/turing/jpg/runner.jpg"]
```



```
rule = {{s, a} -> {A, a, +1}, {s, b} -> {B, b, +1}, {A, a} -> {A, a, +1}, {A, b} -> {A, a, +1},
  {A, " "} -> {A, a, +1}, {B, a} -> {B, b, +1}, {B, b} -> {B, b, +1}, {B, " "} -> {B, b, +1}};
```

```
tape = {" ", a, b, a, b, b, " ", " ", " ", " ", " ", " ", " ", " "};
runner = TuringMachine[rule, {{s, 2}, tape}, 10];
qbracket /@%
{ , a, b, a, b, b, , , , , , }
{{ , [a]s, b, a, b, b, , , , , , },
 { , a, [b]A, a, b, b, , , , , , }, { , a, a, [a]A, b, b, , , , , , },
 { , a, a, a, [b]A, b, , , , , , }, { , a, a, a, a, [b]A, , , , , , },
 { , a, a, a, a, a, [ ]A, , , , , , }, { , a, a, a, a, a, a, [ ]A, , , , , },
 { , a, a, a, a, a, a, a, [ ]A, , , , , }, { , a, a, a, a, a, a, a, a, [ ]A, , , , },
 { , a, a, a, a, a, a, a, a, a, [ ]A, , , , }, { , a, a, a, a, a, a, a, a, a, a, [ ]A, }}}
```

```
turing[runner]
```

