

高知大学教育学部の情報数学のテキスト

文責：高知大学名誉教授 中村 治

チューリング・マシン

チューリング・マシンは計算模型のひとつで計算機を数学的に議論するための、単純化・理想化された仮想機械である。1936年にイギリスの数学者アラン・チューリングの論文「計算可能数について——決定問題への応用」で発表された。

チューリング・マシンは1本のテープと、このテープ上の記号を読み書きするためのヘッドを一つ持つだけの仮想機械である。チューリング・マシンには有限個の「状態」があって、各時刻でそのうち何れか一つの状態をとることができる。

チューリング・マシンのプログラムは次の五つの記号で表される命令のリストで表現されます。一組の命令は

＜現在の状態、読み込んだカラー、書き込むカラー、新しい状態、動く方向＞

の形で表されます。このチューリング・マシンではスペース（または空白）のカラーはBで表現します。また、動く方向はLとRで示します。チューリング・マシンの処理は、次の六つのステップの繰り返しです。

- ヘッドの現在いるセルの記号を読む

- プログラムの全体を調べて、現在の状態である＜現在の状態、読み込んだカラー＞で始まる命令セットを見つけだす。該当する命令セットがなければ、停止する。見つかったら、その命令セットの＜書き込むカラー、新しい状態、動く方向＞を確認する。

- ヘッドの状態を＜新しい状態＞にする。

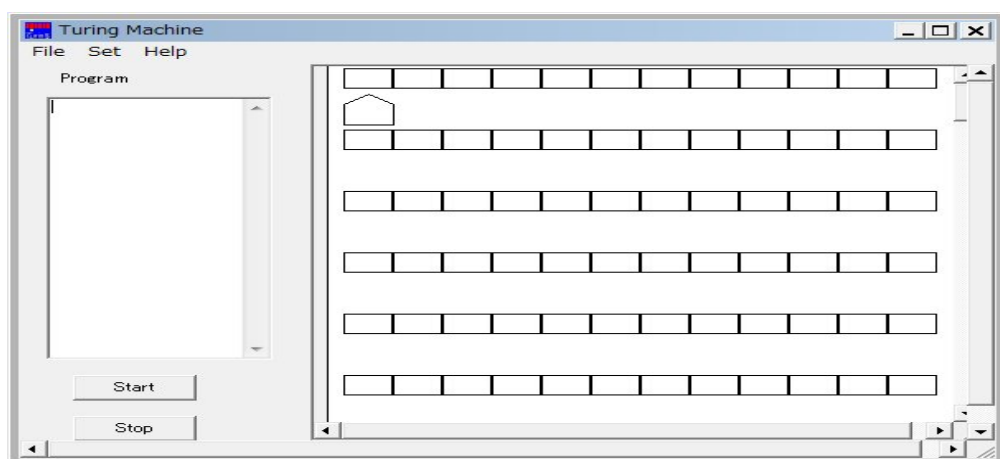
- 現在のセルの内容を＜書き込むカラー＞の記号に変える。

- ＜動く方向＞によって指定された位置にヘッドを移動する。

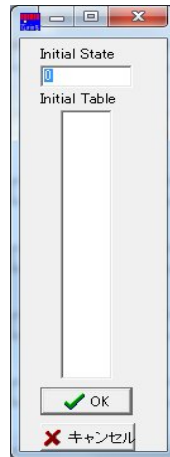
- 最初のステップに戻る。

TURING.EXE でプログラミングしてみましょう。

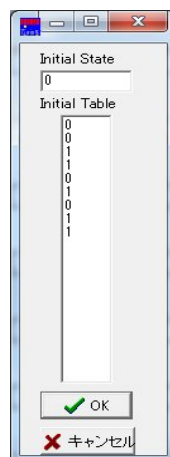
TURING.EXE を立ち上げます。



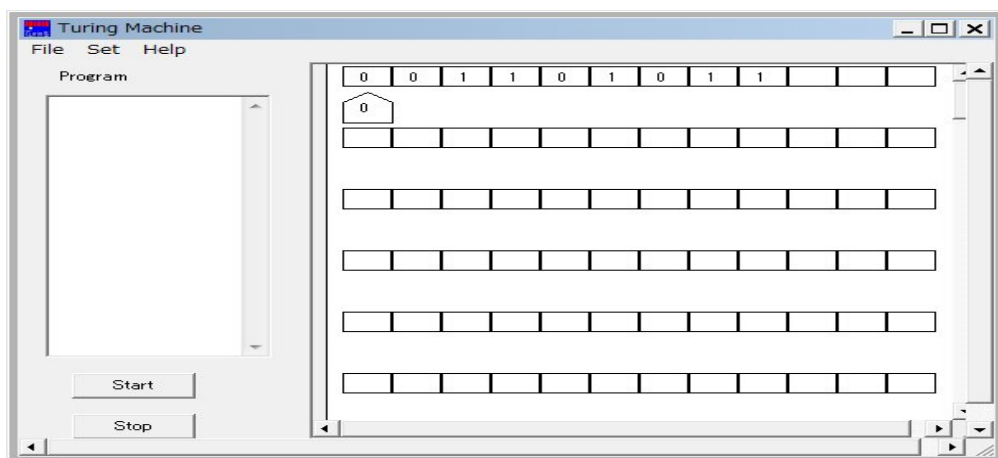
セルとヘッドの初期状態はメニューの set の table で指定します。メニューの set の table をクリックします。



このダイアログボックスで、初期状態と初期データを入力します。通常、初期状態は 0 ですの
で、デフォルトのままで良いです。初期データは、1 行に一文字入力します。例えば、次のように、
001101011 と入力します。



OK のボタンを押すと次のようになります。

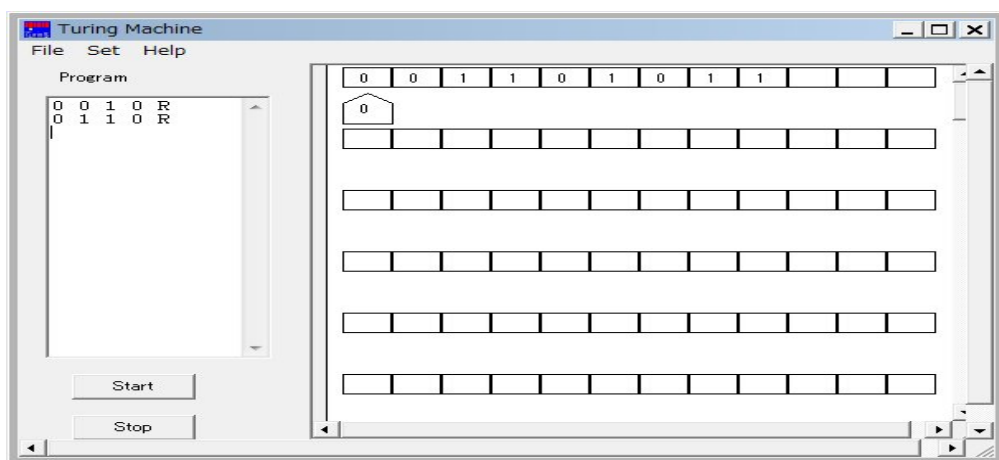


チューリング・マシンのプログラムは、小さな子供の遊んでいる様子をイメージすればいいです。

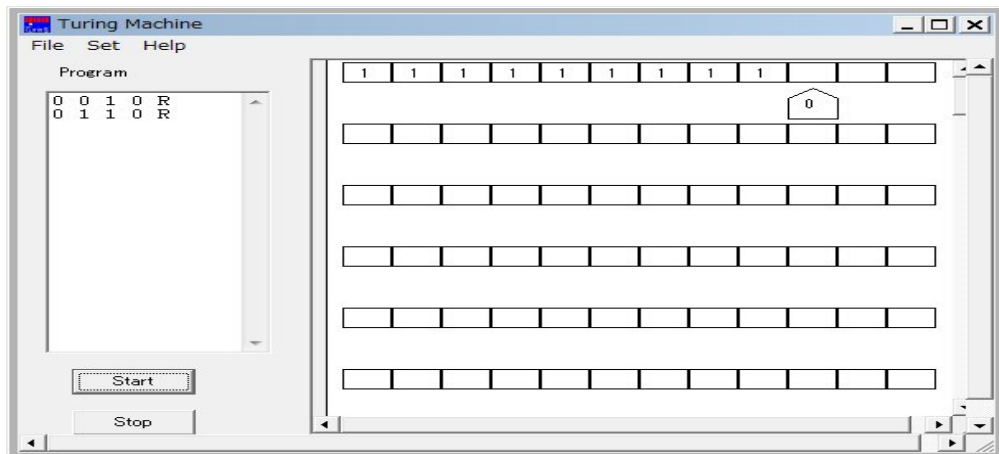
上図のような初期データが与えられたとき 0 をすべて 1 に変えるプログラムを考えて見ます。
 先頭から右に一つずつ調べて、0 があれば 1 に変え、1 であればそのままにします。一つ処理した後も同じことをすればいいので、状態を変える必要はありません。従って、

```
0 0 1 0 R
0 1 1 0 R
```

とプログラミングすればいいです。大して知的なことをしていないので、プログラムも簡単です。
 このプログラムを左側のメモにプログラムを打ち込みます。プログラムは直接編集することも保存することも読み込むこともできます。命令リストの最後は何も書き込まないで下さい。改行だけの行を作らないという意味です。



Start のボタンを押せば、チューリング・マシンが動き出します。



すべての 0 が 1 になりました。
 同様に、0 と 1 をすべて入れ替えるプログラムは、

```
0 0 1 0 R
0 1 0 0 R
```

とプログラミングすればいいです。初期データを入力し直して、実行してみてください。

＜例題＞ SAMPLE DATA : XXXXX が与えられた時、XXXXXX のように、最後に一つ X を追加するプログラムを作れ。

解答例：

```
0 X X 0 R
0 B X 1 R
```

これを

```
0 X X 0 R
0 B X 0 R
```

とすると何が起こるか確かめなさい。

＜例題＞ 例えば SAMPLE DATA : XXXXX が与えられた時、XXXXXXXX のように、最後に2つ X を追加するプログラムを作れ。

解答例：

```
0 X X 0 R
0 B X 1 R
1 B X 2 R
```

これを

```
0 X X 0 R
0 B X 1 R
1 B X 0 L
```

とすると何が起こるか確かめなさい。

＜例題＞ SAMPLE DATA : XXXXX が与えられた時、XxXxXx のように、一つおきに X を x に変えるプログラムを作れ。

解答例：

```
0 X X 1 R
1 X x 0 R
```

＜例題＞ SAMPLE DATA : XXXXXXXX が与えられた時、XXxXXxX のように、二つおきに X を x に変えるプログラムを作れ。

解答例：

```
0 X X 1 R
1 X X 2 R
2 X x 0 R
```

＜例題＞ SAMPLE DATA : XXX が与えられた時、BBBBXXX のように、X 達を移動するプログラムを作れ。ここで、B は空白を示しています。

解答例：

```

0 X B 1 R
1 X X 1 R
1 B B 2 R
2 B X 3 L
3 B B 4 L
4 X X 4 L
4 B B 0 R
2 X X 2 R
3 X X 3 L

```

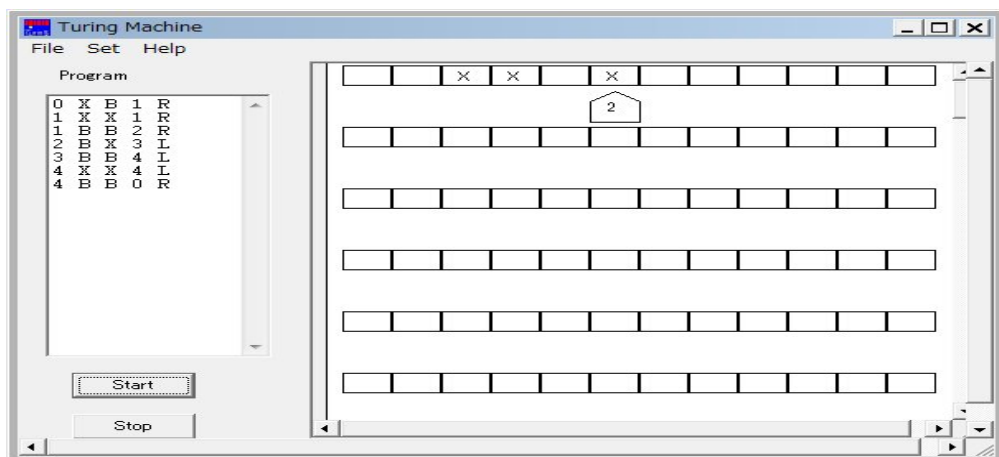
これは次のように考えればよい。初期状態 0 で X を読めば、空白 (B) を書き込み、「X を抱えて置く場所を見つけるために区切りの空白を見つけようと右に移動している状態 1」として、右に移動する。状態 1 で X を読めば、X を読み飛ばして、同じ状態のまま右に移動する。状態 1 で B (空白) を読めば、B (空白) を読み飛ばして、「X を抱えて置く場所を見つけるために最初の空白を見つけようと右に移動している状態 2」として、右に移動する。状態 2 で B (空白) を読めば、X を書き込み、「次の X を見つけるために区切りの空白を見つけようと左に移動している状態 3」として、左に移動する。状態 3 で B (空白) を読めば、B (空白) を読み飛ばして、「次の X を見つけるために最初の空白を見つけようと左に移動している状態 4」として、左に移動する。状態 4 で X を読めば、X を読み飛ばして、同じ状態のまま左移動する。状態 4 で B (空白) を読めば、B (空白) を読み飛ばして、状態を 0 にして右に移動する。

```

0 X B 1 R
1 X X 1 R
1 B B 2 R
2 B X 3 L
3 B B 4 L
4 X X 4 L
4 B B 0 R

```

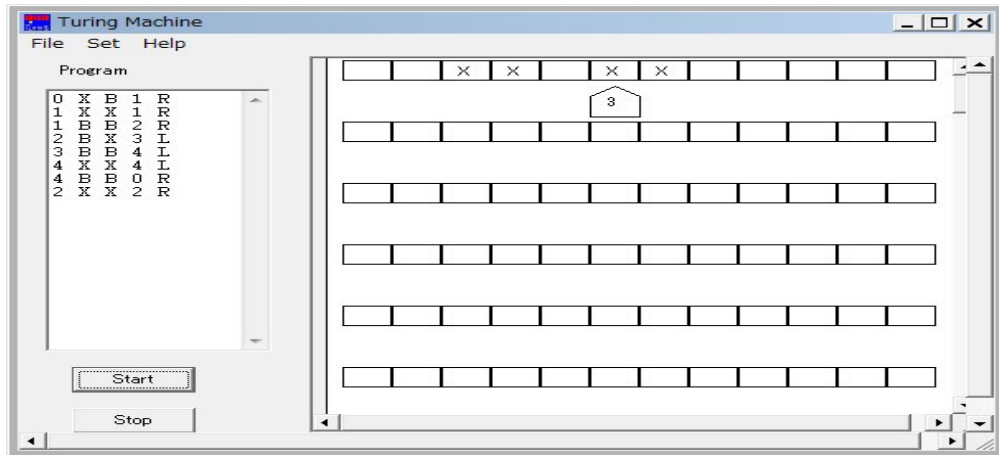
これでプログラムが出来たと思って実行してみると



と、「X を抱えて置く場所を見つけるために最初の空白を見つけようと右に移動している状態 2」で X を読んだところで停止します。すでに移動した X がある場合の考察を忘れていました。状態 2 で X を読めば、X を読み飛ばして、同じ状態のまま右移動するという命令

2 X X 2 R

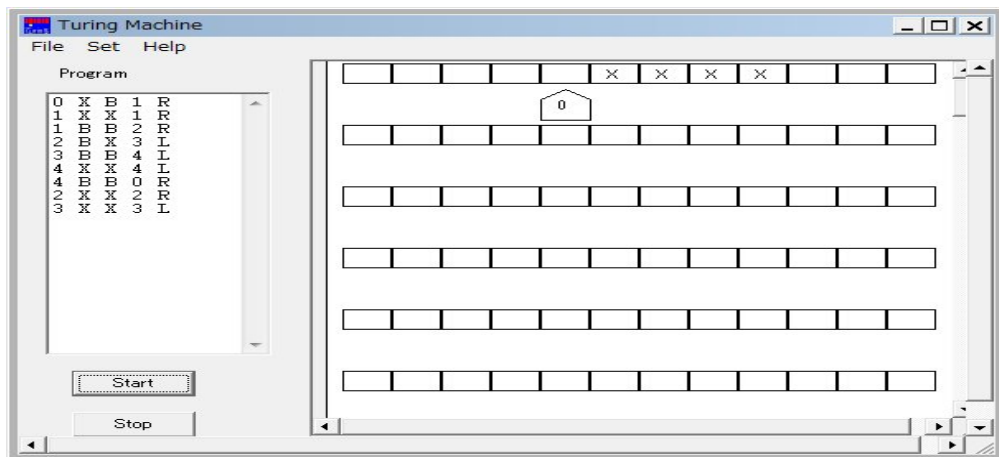
を追加して、再び実行します。



今度は、「次の X をを見つけるために区切りの空白を見つけようと左に移動している状態 3」で、X を読んだところで停止します。ここでも、すでに移動した X がある場合の考察を忘れていました。

3 X X 3 L

を追加して、再び実行します。



今度は旨くいきました。このようにプログラムを作るときは、最初からすべてが完全に出来ることはまずないです。何度も何度も失敗し、あらゆる失敗を経験した後、プログラムが作れるようになります。以下で学ぶどのプログラミング言語を使う場合でも、プログラミングが出来るようになるかどうかは、何度も何度も失敗し、あらゆる失敗を経験するまで我慢出来るかどうかです。

問題：SAMPLE DATA : XXX が与えられた時、BBBBXXXXXX のように、任意の X の列の X の個数の 2 倍の X からなるの列を描くプログラムを作れ。ここで、B は空白を示しています。

例えば SAMPLE DATA : BXXXX が与えられた時、

0 B B 1 R

1 X X 1 R

1 B B 2 L

2 X X 2 L
2 B B 1 R

のプログラムを実行すると何が起こるか確かめなさい。ここで、B は空白を示しています。

例えば SAMPLE DATA : BXXXX が与えられた時、

0 B B 1 R
1 X X 1 R
1 B X 2 L
2 X X 2 L
2 B B 1 R

のプログラムを実行すると何が起こるか確かめなさい。ここで、B は空白を示しています。

<例題> SAMPLE DATA : ()(()) や (()() が与えられた時、括弧がうまく対応しているかどうか判定できるプログラムを作れ。

解答例：

0 (Y 1 R
0 X X 0 R
0) Z 3 R
0 B B 5 R
1 ((1 R
1) X 2 L
1 X X 1 R
1 B B 4 R
2 (X 1 R
2 X X 2 L
2 Y Z 0 R
2 Z Z 3 R
5 B 0 6 R
6 B K 7 R
3 ((3 R
3)) 3 R
3 B B 8 R
8 B N 9 R
9 B 0 10 R
4 B B 8 R

8 X X 8 L
 9 X X 9 L
 7 0 X 10 R
 7 1 X 11 R
 8 0 X 11 R
 8 1 X 12 R
 9 0 X 12 R
 9 1 X 13 R
 10 0 0 10 R
 10 1 1 10 R
 10 + + 10 R
 10 = = 10 R
 10 X X 10 R
 10 Y Y 10 R
 11 0 0 11 R
 11 1 1 11 R
 11 + + 11 R
 11 = = 11 R
 11 X X 11 R
 11 Y Y 11 R
 12 0 0 12 R
 12 1 1 12 R
 12 + + 12 R
 12 = = 12 R
 12 X X 12 R
 12 Y Y 12 R
 13 0 0 13 R
 13 1 1 13 R
 13 + + 13 R
 13 = = 13 R
 13 X X 13 R
 13 Y Y 13 R
 10 B B 14 L
 11 B B 15 L
 12 B B 16 L
 13 B B 17 L
 14 0 B 18 R
 14 1 B 19 R
 15 0 B 20 R
 15 1 B 21 R
 16 0 B 22 R
 16 1 B 23 R
 17 0 B 24 R

17 1 B 25 R
 18 B 0 14 L
 19 B 1 14 L
 20 B 0 15 L
 21 B 1 15 L
 22 B 0 16 L
 23 B 1 16 L
 24 B 0 17 L
 25 B 1 17 L
 14 B B 14 L
 15 B B 15 L
 16 B B 16 L
 17 B B 17 L
 14 = = 26 R
 15 = = 27 R
 16 = = 28 R
 17 = = 29 R
 26 B 0 2 L
 27 B 1 2 L
 28 B 0 3 L
 29 B 1 3 L
 2 = = 2 L
 3 = = 3 L
 2 Y Y 2 L
 3 Y Y 3 L
 2 + + 30 L
 3 + + 31 L
 30 X X 30 L
 31 X X 31 L
 30 Z X 32 R
 30 W X 33 R
 31 Z X 33 R
 31 W X 34 R
 7 Z X 32 R
 7 W X 33 R
 8 Z X 33 R
 8 W X 34 R
 9 Z X 34 R
 9 W X 35 R
 32 X X 32 R
 32 + + 32 R
 32 Y Y 32 R
 32 = = 32 R

32 0 0 32 R
 32 1 1 32 R
 33 X X 33 R
 33 + + 33 R
 33 Y Y 33 R
 33 = = 33 R
 33 0 0 33 R
 33 1 1 33 R
 34 X X 34 R
 34 + + 34 R
 34 Y Y 34 R
 34 = = 34 R
 34 0 0 34 R
 34 1 1 34 R
 35 X X 35 R
 35 + + 35 R
 35 Y Y 35 R
 35 = = 35 R
 35 0 0 35 R
 35 1 1 35 R
 32 B B 36 L
 33 B B 37 L
 34 B B 38 L
 35 B B 39 L
 36 0 B 40 R
 36 1 B 41 R
 37 0 B 42 R
 37 1 B 43 R
 38 0 B 44 R
 38 1 B 45 R
 39 0 B 46 R
 39 1 B 47 R
 40 B 0 36 L
 41 B 1 36 L
 42 B 0 37 L
 43 B 1 37 L
 44 B 0 38 L
 45 B 1 38 L
 46 B 0 39 L
 47 B 1 39 L
 36 B B 36 L
 37 B B 37 L
 38 B B 38 L

39 B B 39 L
 36 = = 48 R
 37 = = 49 R
 38 = = 50 R
 39 = = 51 R
 48 B 0 -1 R
 49 B 1 -1 R
 50 B 0 52 R
 51 B 1 52 R
 52 0 0 52 R
 52 1 1 52 R
 52 B B 53 L
 53 0 B 54 R
 53 1 B 55 R
 54 B 0 53 L
 55 B 1 53 L
 53 B B 53 L
 53 = = 56 R
 56 B 1 -1 R
 30 0 X 57 R
 30 1 X 58 R
 31 0 X 58 R
 31 1 X 59 R
 57 X X 57 R
 57 + + 57 R
 57 Y Y 57 R
 57 = = 57 R
 57 0 0 57 R
 57 1 1 57 R
 58 X X 58 R
 58 + + 58 R
 58 Y Y 58 R
 58 = = 58 R
 58 0 0 58 R
 58 1 1 58 R
 59 X X 59 R
 59 + + 59 R
 59 Y Y 59 R
 59 = = 59 R
 59 0 0 59 R
 59 1 1 59 R
 57 B B 60 L
 58 B B 61 L

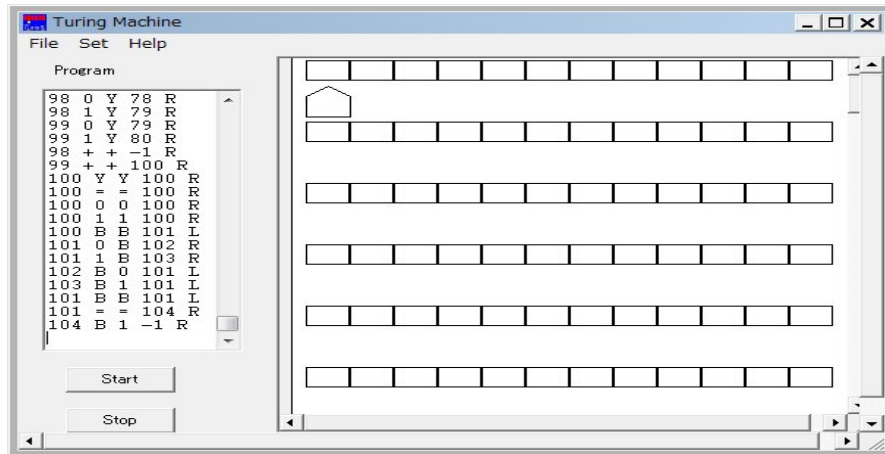
59 B B 62 L
 60 0 B 63 R
 60 1 B 64 R
 61 0 B 65 R
 61 1 B 66 R
 62 0 B 67 R
 62 1 B 68 R
 63 B 0 60 L
 64 B 1 60 L
 65 B 0 61 L
 66 B 1 61 L
 67 B 0 62 L
 68 B 1 62 L
 60 B B 60 L
 61 B B 61 L
 62 B B 62 L
 60 = = 69 R
 61 = = 70 R
 62 = = 71 R
 69 B 0 2 L
 70 B 1 2 L
 71 B 0 3 L
 4 0 0 72 L
 4 1 1 72 L
 5 0 0 73 L
 5 1 1 73 L
 6 0 0 74 L
 6 1 1 74 L
 72 0 0 72 L
 72 1 1 72 L
 73 0 0 73 L
 73 1 1 73 L
 74 0 0 74 L
 74 1 1 74 L
 72 + + 75 L
 73 + + 76 L
 74 + + 77 L
 75 X X 75 L
 76 X X 76 L
 77 X X 77 L
 75 0 X 10 R
 75 1 X 11 R
 76 0 X 11 R

76 1 X 12 R
 77 0 X 12 R
 77 1 X 13 R
 75 Z X 78 R
 75 W X 79 R
 76 Z X 79 R
 76 W X 80 R
 77 Z X 80 R
 77 W X 81 R
 78 X X 78 R
 78 + + 78 R
 78 0 0 78 R
 78 1 1 78 R
 78 Y Y 78 R
 78 = = 78 R
 79 X X 79 R
 79 + + 79 R
 79 0 0 79 R
 79 1 1 79 R
 79 Y Y 79 R
 79 = = 79 R
 80 X X 80 R
 80 + + 80 R
 80 0 0 80 R
 80 1 1 80 R
 80 Y Y 80 R
 80 = = 80 R
 81 X X 81 R
 81 + + 81 R
 81 0 0 81 R
 81 1 1 81 R
 81 Y Y 81 R
 81 = = 81 R
 78 B B 82 L
 79 B B 83 L
 80 B B 84 L
 81 B B 85 L
 82 0 B 86 R
 82 1 B 87 R
 83 0 B 88 R
 83 1 B 89 R
 84 0 B 90 R
 84 1 B 91 R

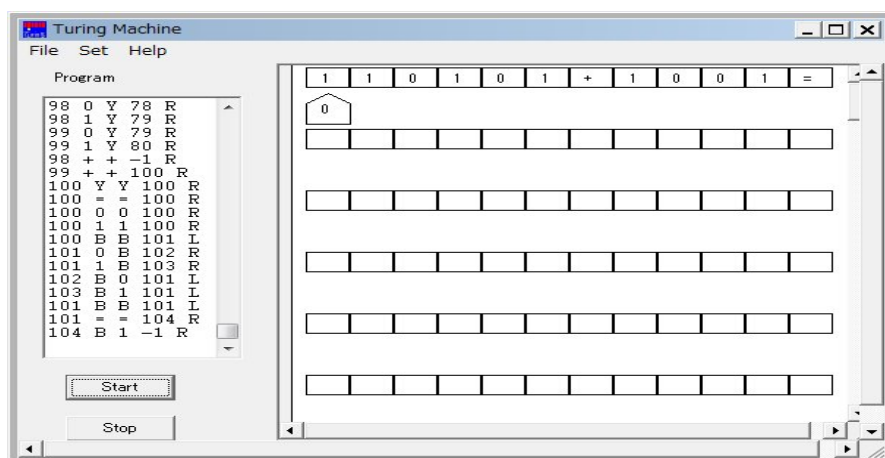
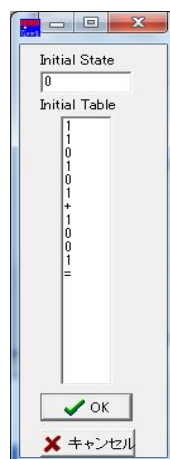
85 0 B 92 R
 85 1 B 93 R
 86 B 0 82 L
 87 B 1 82 L
 88 B 0 83 L
 89 B 1 83 L
 90 B 0 84 L
 91 B 1 84 L
 92 B 0 85 L
 93 B 1 85 L
 82 B B 82 L
 83 B B 83 L
 84 B B 84 L
 85 B B 85 L
 82 = = 94 R
 83 = = 95 R
 84 = = 96 R
 85 = = 97 R
 94 B 0 98 L
 95 B 1 98 L
 96 B 0 99 L
 97 B 1 99 L
 98 = = 98 L
 99 = = 99 L
 98 Y Y 98 L
 99 Y Y 99 L
 98 0 Y 78 R
 98 1 Y 79 R
 99 0 Y 79 R
 99 1 Y 80 R
 98 + + -1 R
 99 + + 100 R
 100 Y Y 100 R
 100 = = 100 R
 100 0 0 100 R
 100 1 1 100 R
 100 B B 101 L
 101 0 B 102 R
 101 1 B 103 R
 102 B 0 101 L
 103 B 1 101 L
 101 B B 101 L
 101 = = 104 R

104 B 1 -1 R

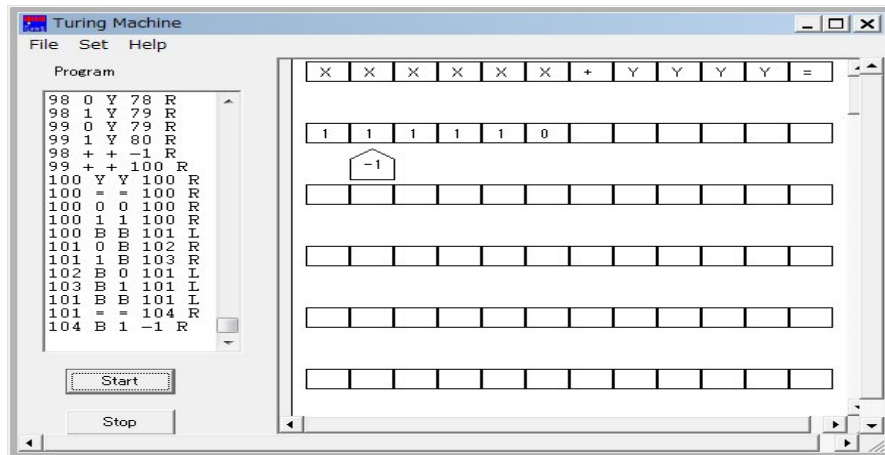
このプログラムをコピーします。



初期データをセットします。

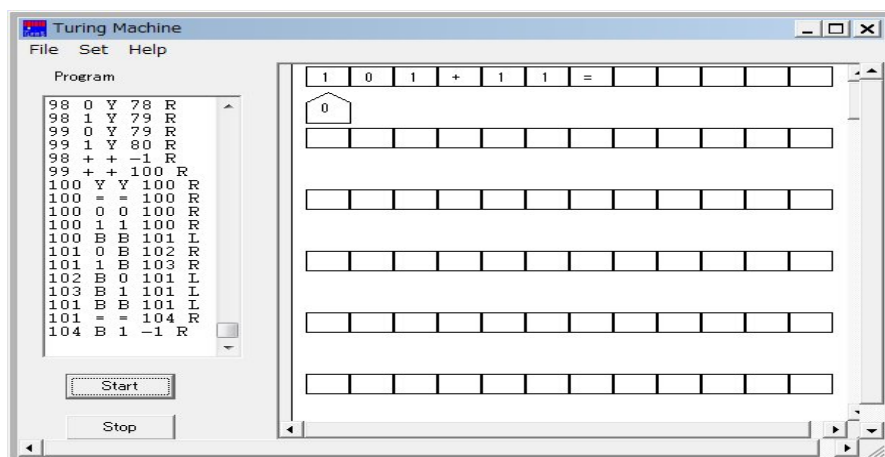
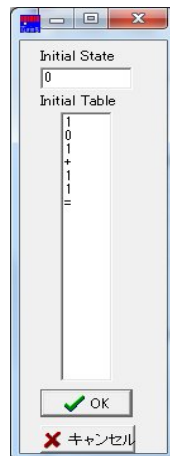


実行すると

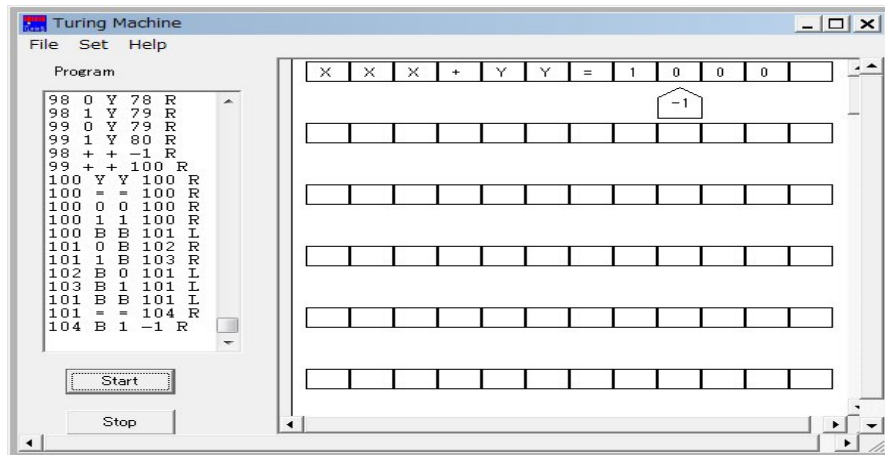


となります。

分かり易い別の例をやってみます。初期データをセットします。



実行すると



となります。

このように数値計算も努力すればプログラミングできます。しかし複雑なプログラムを作るのは大変です。私は大きな模造紙にプログラムの流れを描いて、チューリング・マシンの動きを図示しながら、丸一日かかってこのプログラムを作り上げました。チューリング・マシンは実際にプログラミングする機械ではなく、計算機の原理を示す機械です。

レポート問題： 2進数の2の補数を計算するプログラムを作れ。

レポート問題： SAMPLE DATA : XYYXYXXY が与えられた時、BBBBBBBBB XYYXYXXYのように、X,Y の任意の列を移動するプログラムを作れ。ここで、B は空白を示しています。

次は現実の計算機を考察します。