

8. Turing 機械入門(2)

8.3. チューリング機械のプログラム技法

- 基本テクニック

8.4. 基本チューリング機械の拡張

- 自然な拡張... 言語は変化しない

プログラミングを
容易にする

8.5. 制限されたチューリング機械

- 自然な制限... 言語は変化しない

TMの扱いを容
易にする

8.6. チューリング機械とコンピュータ

- チューリング機械は万能性を持ち、通常のコンピュータと同じ計算能力を持つこと

8. Turing Machine (2)

8.3. Programming techniques for TM

- Basic techniques

Easy for programming

8.4. Extended TM

- Natural extensions... language does not change

8.5. Restricted TM

Easy for handling TM

- Natural restrictions... language does not change

8.6. Turing machine and Computer

- Turing machine has **universality**, and has the **same computation power** as usual computer

8.3. プログラミング技法

基本テクニック

1. 状態に「記憶」する
2. テープのトラックを増やす
3. サブルーチン

8.3. Programming techniques

Basic techniques

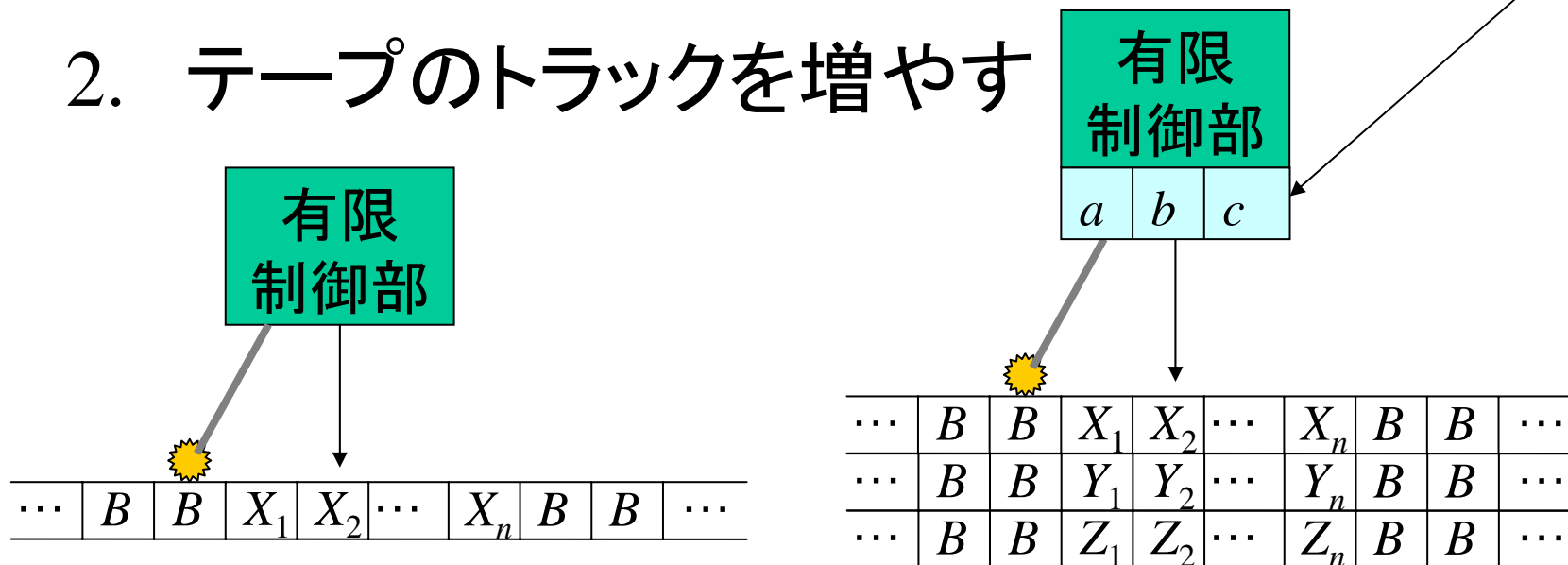
1. States can be used as memories
2. Increasing the number of tracks in a tape
3. Subroutines

8.3. プログラミング技法

基本テクニック

1. 状態に「記憶」する
2. テープのトラックを増やす

1. 入力長と無関係の「有限の値」を「状態」で覚える
(q, a, b, c)と書けばよい



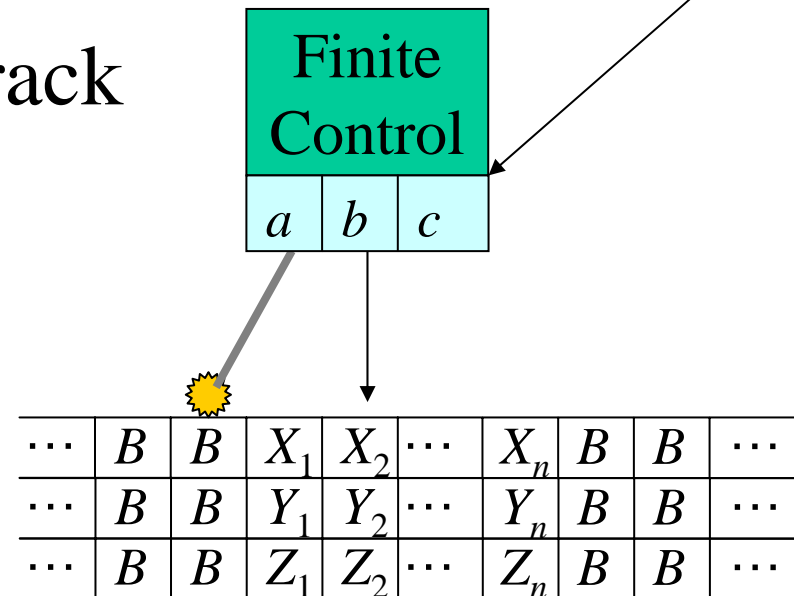
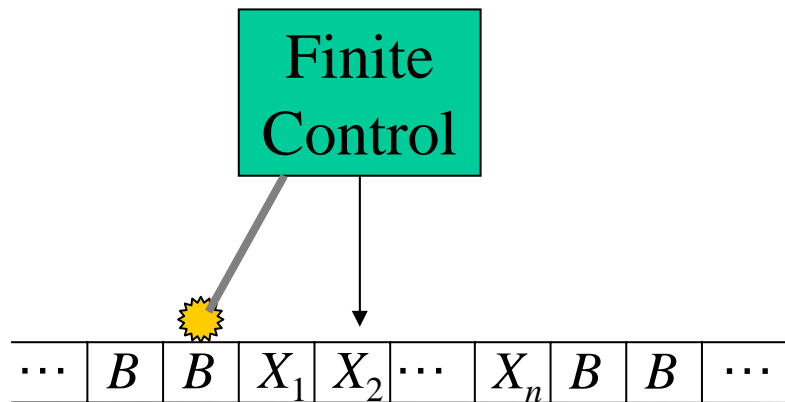
2. テープアルファベットを3つ組にする。 (X_i, Y_i, Z_i) と書けばよい

8.3. Programming techniques

Basic Techniques

1. Store using states
2. Split the tape to multi-track

1. 'Finite values' can be stored by states, which can be denoted by, e.g., (q, a, b, c)



2. Multi-track can be realized by changing an alphabet to, e.g., 3-tuple (X_i, Y_i, Z_i) .

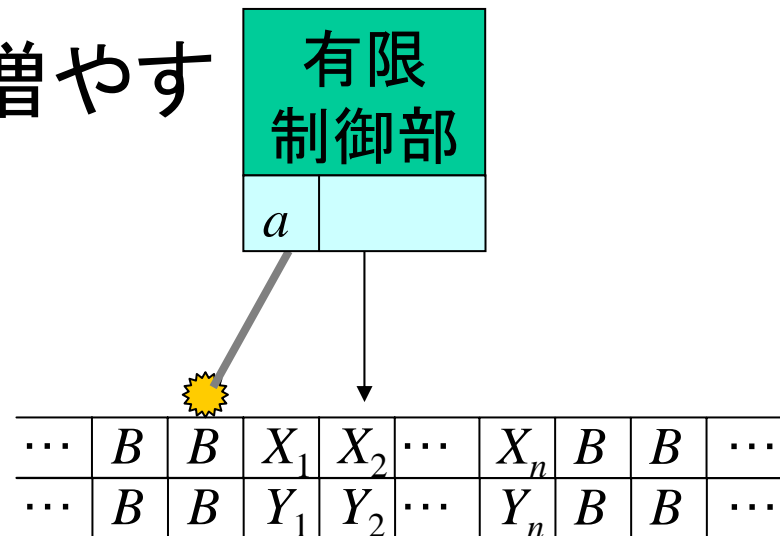
8.3. プログラミング技法

基本テクニック

1. 状態に「記憶」する
2. テープのトラックを増やす

例) 前回の回文なら...

1. 状態で0/1を記憶
2. テープのYにチェックを入れればXを消す必要はない



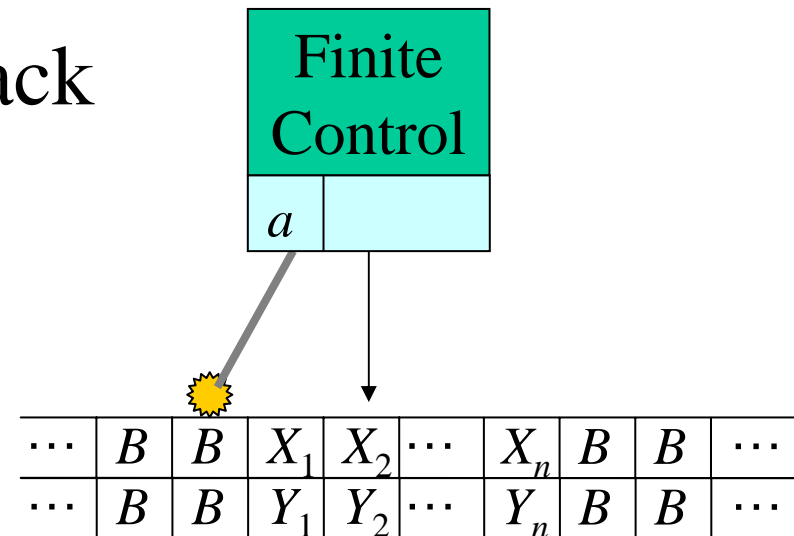
8.3. Programming techniques

Basic Techniques

1. Store using states
2. Split the tape to multi-track

Ex) For the palindrome...

1. 'state' stores 0/1
2. no need to delete X by checking Y on the other track.



8.3. プログラミング技法

基本テクニック

3. サブルーチン(詳細は略)

- あるまとまった処理を行うTMプログラムを何度も再利用する方法...状態遷移図上でコピーを作つて、埋め込む

8.3. Programming techniques

Basic Techniques

3. Subroutine (details are omitted)

- Re-use a TM program that performs some unit computation ... make many copies and embed them into the transition diagram.

8.4.基本チューリング機械の拡張

自然な拡張...言語としては変わらない

1. 多テープ TM
2. 非決定性 TM

決定性: 遷移先は一意に決定

非決定性: 遷移先は複数存在。

どれか一つでも受理なら受理。

8.4. Extended Turing Machine

Natural **Extension**...that does not change the class of languages

1. Multi-tape TM
2. Nondeterministic TM

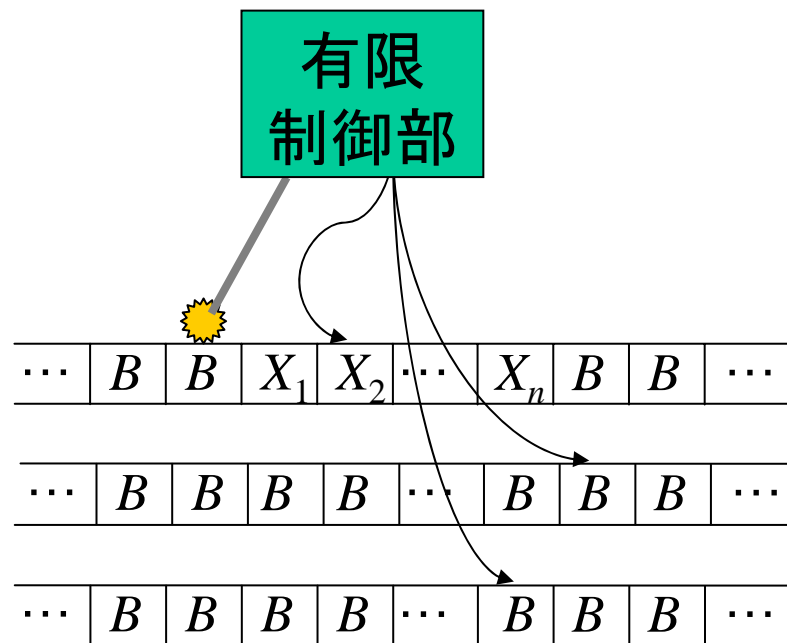
Deterministic: the next state is uniquely determined

Nondeterministic: there can be many next states. TM accepts the input if one of possible state accepts.

8.4.基本チューリング機械の拡張

自然な拡張...言語としては変わらない

1. 多テープ TM

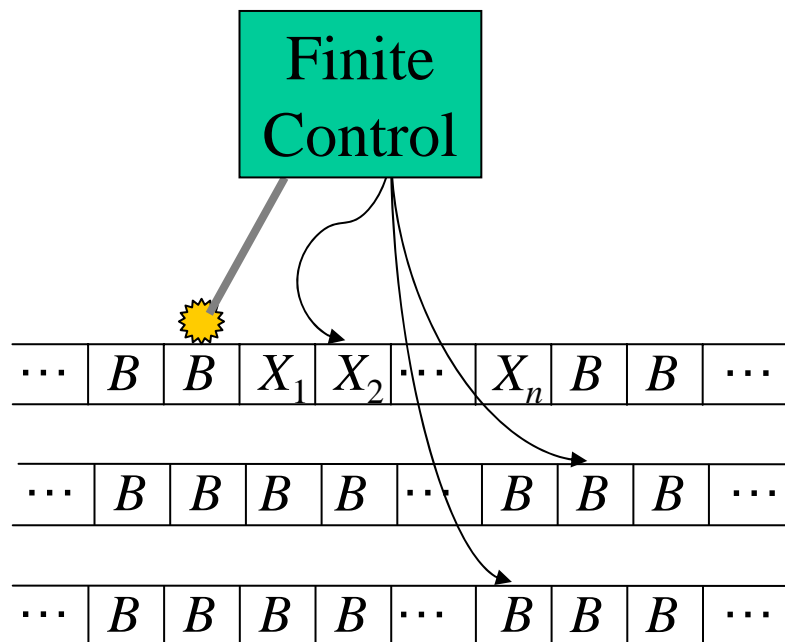


- ヘッドが各テープごとに独立に読み書き
- 入力テープ以外は最初はすべて B

8.4. Extended Turing Machine

Natural **Extension**...that does not change the class of languages

1. Multi-tape TM

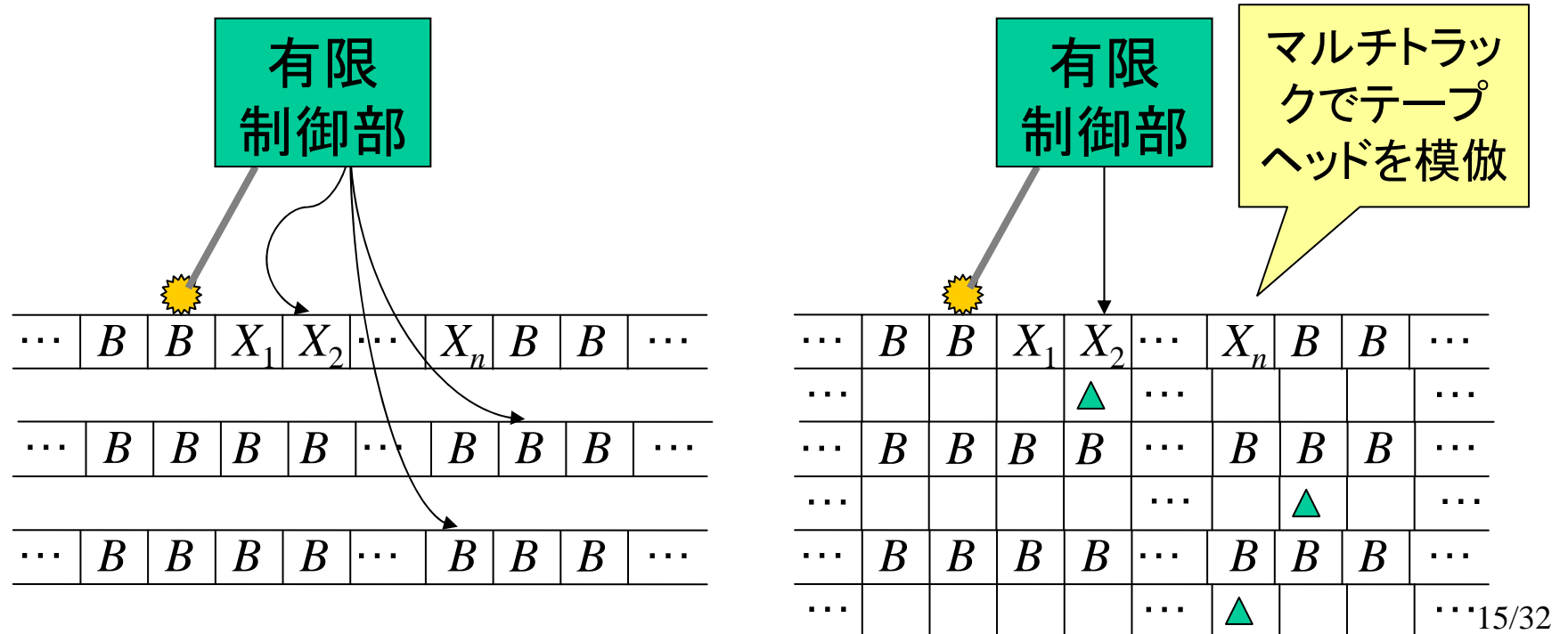


- Each tape is read/written by distinct heads independently.
- Except input tape, all tapes are filled by B .

8.4.基本チューリング機械の拡張

自然な拡張...言語としては変わらない

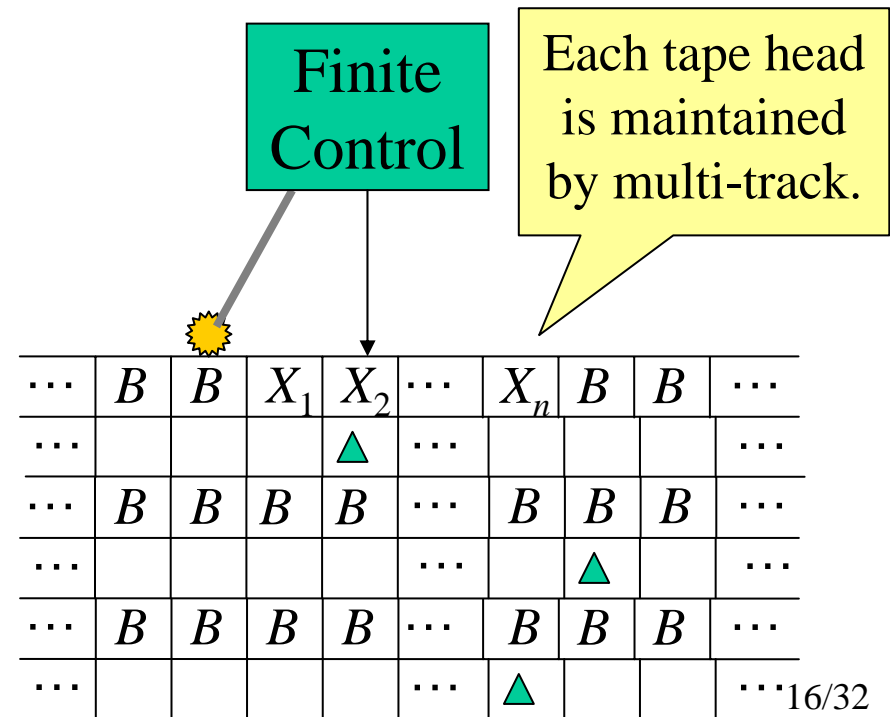
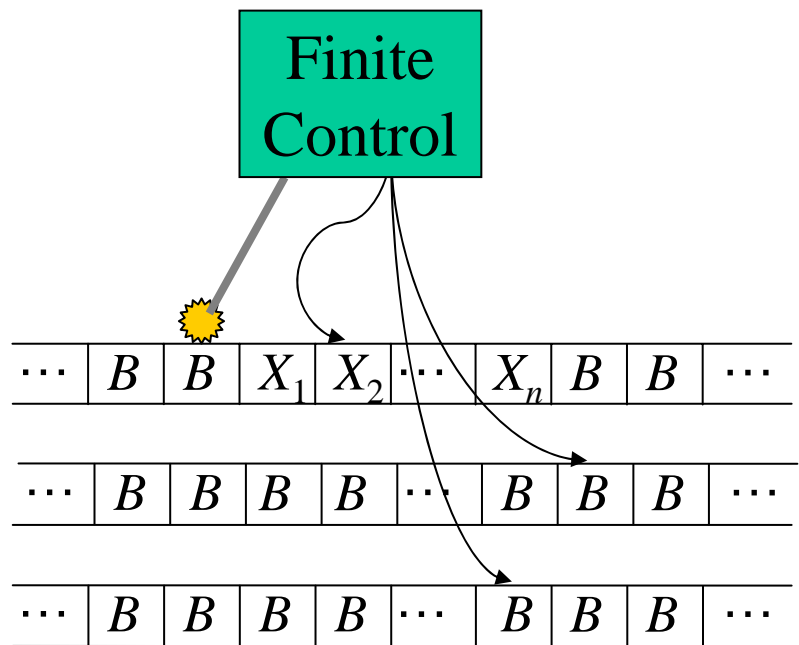
1. 多テープ TM を1テープ TM で模倣(概要)



8.4. Extended Turing Machine

Natural **Extension**...that does not change the class of languages

1. Multi-tape TM can be simulated by a one-tape TM (Sketch)



8.4.基本チューリング機械の拡張


自然な拡張...言語としては変わらない

2. 非決定性 TM を決定性の TM で模倣(概要)

非決定性 TM:

- 「次の状態」が複数ある
- 一つでも受理状態にたどり着く遷移の列が存在すれば受理

決定性 TM による模倣:



計算時間は爆発的に増加

- 複数の「次の状態」のどれを選んだか、別テープに記録しておく
- 可能な選択肢をすべて順番にチェックして、一つでも受理状態にたどり着く遷移の列が存在すれば受理

8.4. Extended Turing Machine

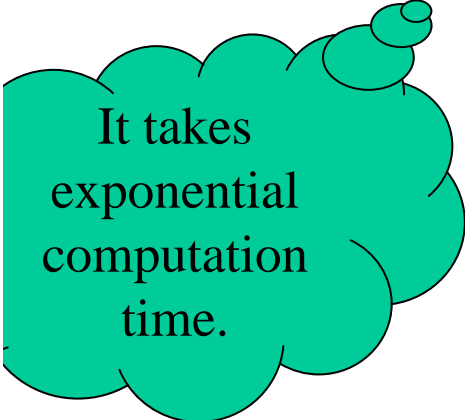
Natural **Extension**...that does not change the class of languages

2. Nondeterministic TM can be simulated by a deterministic TM (Sketch)

Nondeterministic TM:

- Many 'next states' exist
- It accepts if one of computations reaches to an accepting state.

Simulation by a deterministic TM:



It takes exponential computation time.

- Record on the other tape the sequence of 'next states' nondeterministic TM chosen
- Check all possible choices, and accept if at least one computation reaches to an accepting state.

8.5. 制限されたチューリング機械

自然な制限...言語としては変わらない

1. 半無限テープを持つ TM
 - 一方だけ無限長で、他方には「端」がある
2. テープの代わりに複数のスタックを持つ TM
 - 2つのスタックを持てば十分!!
3. テープの代わりにカウンタを持つ TM(省略)
 - カウンタ1つ=PDAと同じ能力=CFL
 - カウンタを2つ持てば十分!!

8.5. Restricted Turing Machine

Natural **Restriction**...that does not change the class of languages

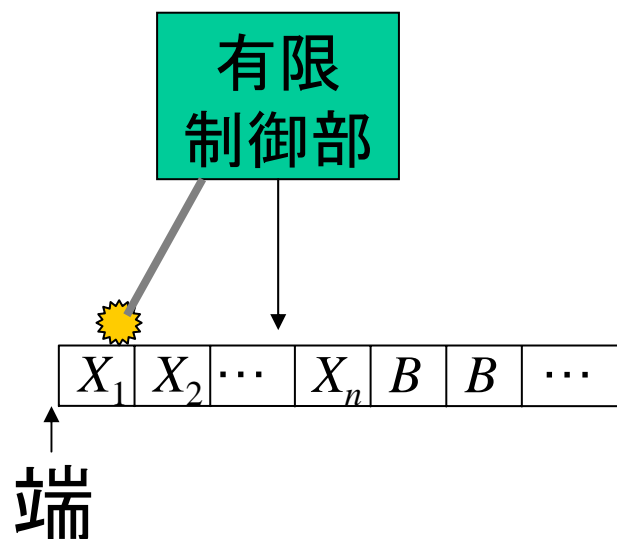
1. TM with semi-infinite tape
 - the tape has the leftmost cell.
2. TM with stacks instead of a tape
 - Two stacks are sufficient!!
3. TM with counter instead of a tape (Omitted)
 - One counter = PDA = CFL
 - Two counters are sufficient!!

8.5. 制限されたチューリング機械

自然な制限... 言語としては変わらない

1. 半無限テープを持つ TM

- 一方だけ無限長で、他方には「端」がある



★ 言語を受理する能力は通常の TM と変わらない。

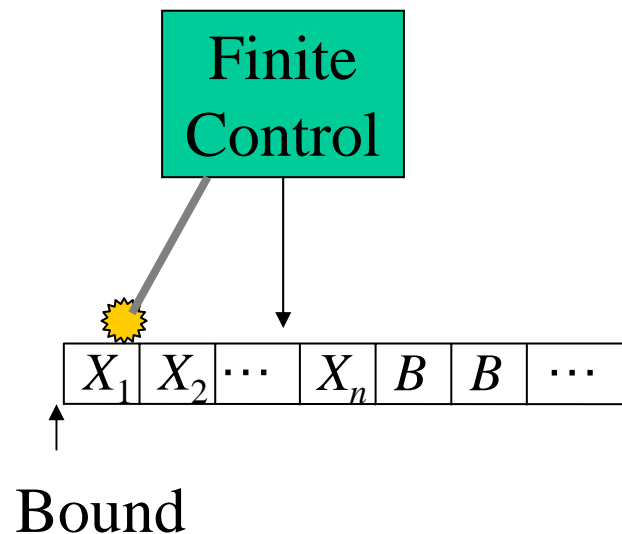
★ 通常のテープを持つ TM を模倣する能力がある。

8.5. Restricted Turing Machine

Natural **Restriction**...that does not change the class of languages

1. TM with semi-infinite tape

- There are no cells on the left of the initial position.



★ It can accept **the same language** as usual TM.

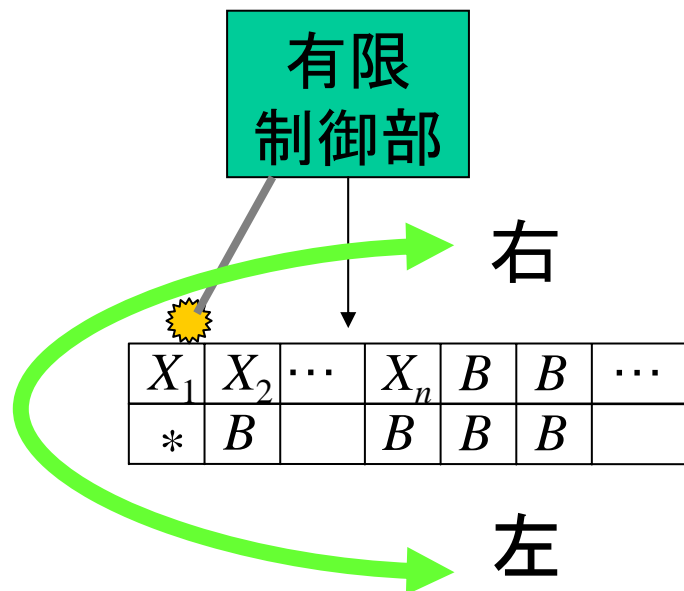
★ It can simulate the usual TM.

8.5. 制限されたチューリング機械

自然な**制限**... 言語としては変わらない

1. 半無限テープを持つ TM

- 一方だけ無限長で、他方には「端」がある



★ 通常のテープを持つ TM を模倣する能力がある。

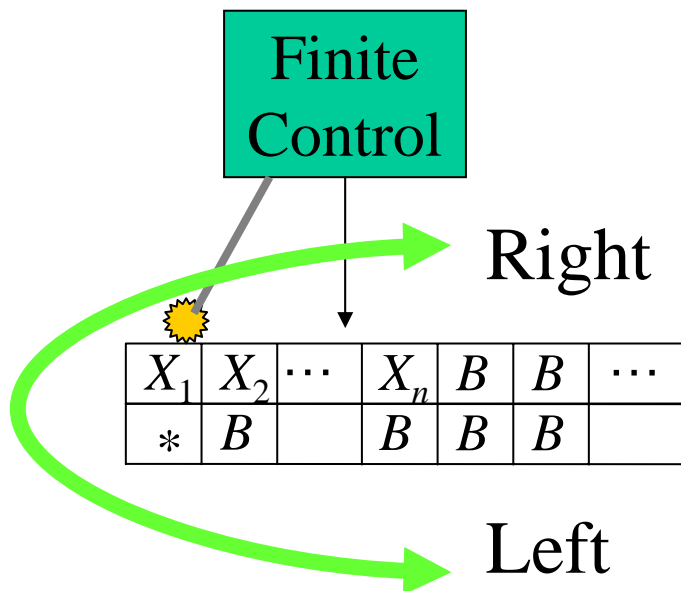
⇒ テープを2トラックと見なして、通常のテープを折りたたんで模倣する。

8.5. Restricted Turing Machine

Natural **Restriction**...that does not change the class of languages

1. TM with semi-infinite tape

- There are no cells on the left of the initial position.



★ It can accept **the same language** as usual TM.

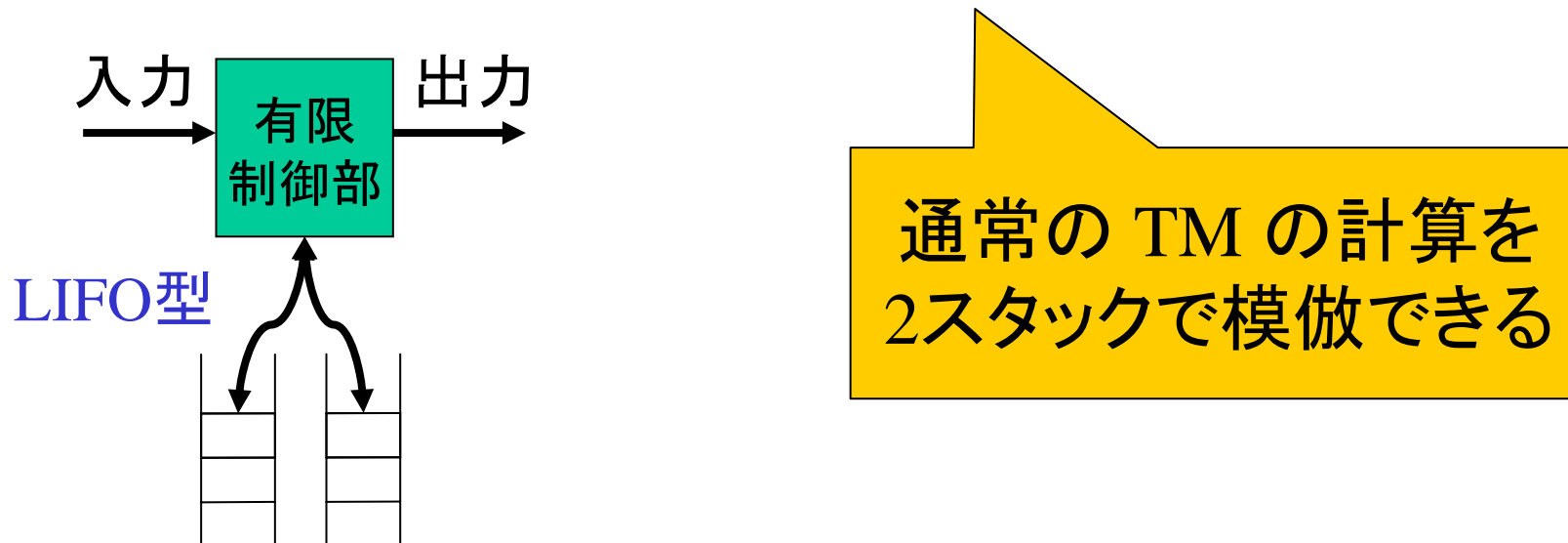
⇒ Using two track tape, simulate the usual tape that has no bounds on both sides.

8.5. 制限されたチューリング機械

自然な制限... 言語としては変わらない

2. テープの代わりに複数のスタックを持つ TM

- 1つのスタック...PDA
- 2つのスタック...TM と同性能

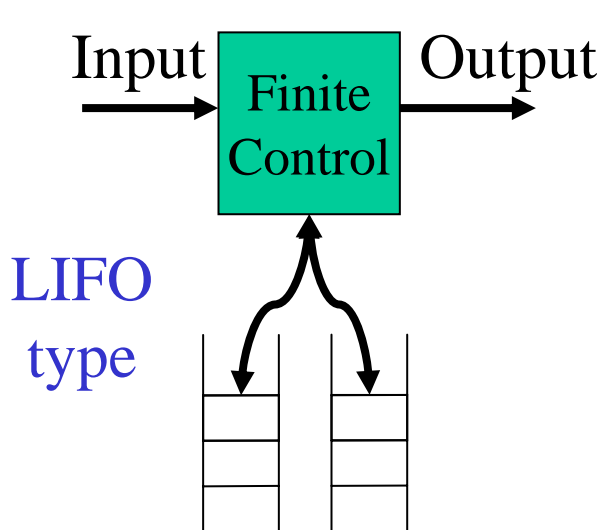


8.5. Restricted Turing Machine

Natural **Restriction**...that does not change the class of languages

2. TM with stacks instead of a tape

- 1 stack...PDA
- 2 stacks... Equivalent to the usual TM

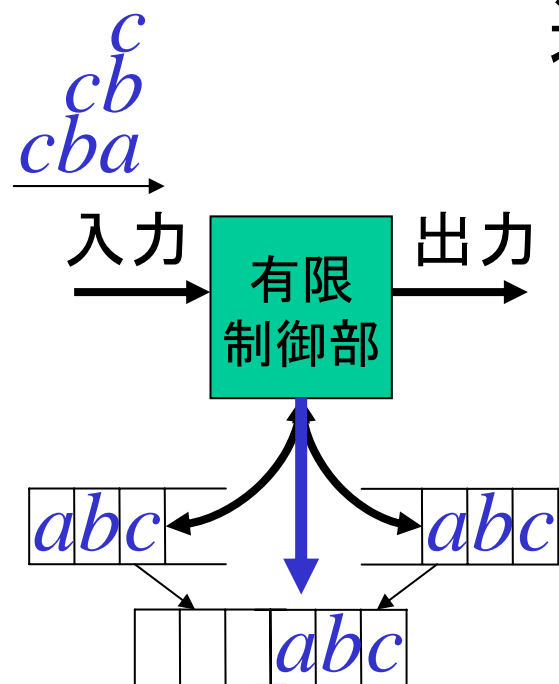


2 stacks are enough to simulate the usual TM.

8.5. 制限されたチューリング機械

自然な制限... 言語としては変わらない

2. テープの代わりに2つのスタックを持つ TM



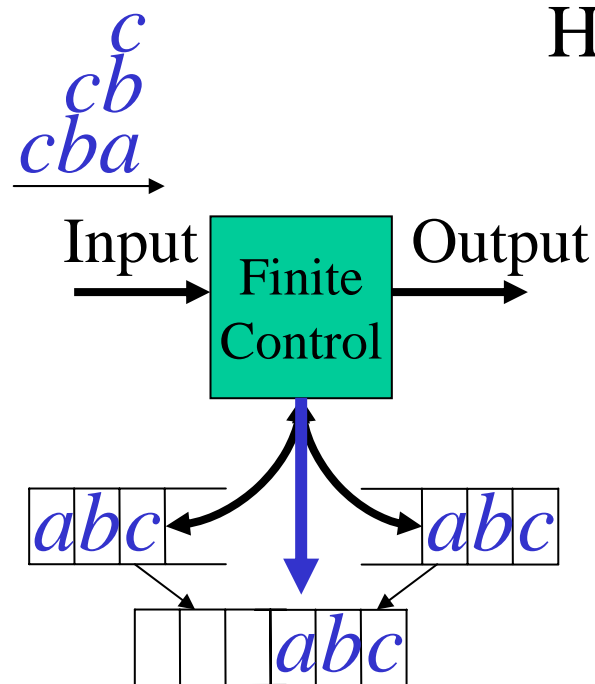
通常の TM の模倣方法(概要):

1. 入力データをすべてスタック1につむ
(入力が逆順に並ぶ)
2. スタック2にコピー
(入力が正順に並ぶ)
3. 二つのスタックを仮想的につないで
一つのテープと見なす

8.5. Restricted Turing Machine

Natural **Restriction**...that does not change the class of languages

2. TM with stacks instead of a tape



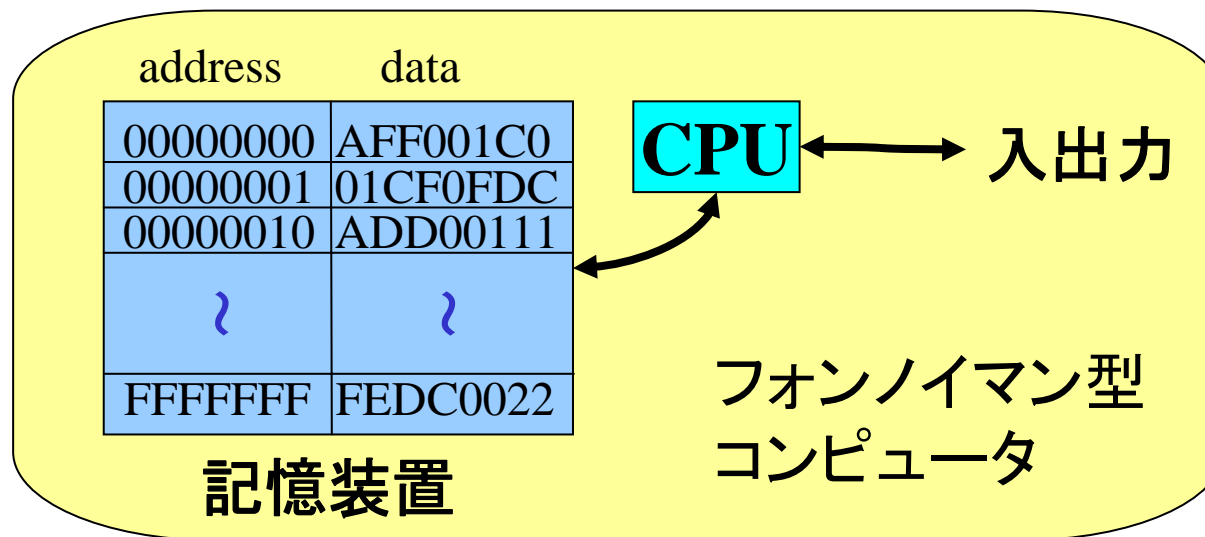
How to simulate usual TM (Sketch):

1. Push all input data to the stack 1 (hence order is reversed)
2. Copy them to the stack 2 (order is turned to usual)
3. Regard two stacks as one tape by joining them.

8.6. Turing 機械とコンピュータ

- チューリング機械は**万能性**を持ち、通常のコンピュータと同じ**計算能力**を持つこと

‘通常のコンピュータ’ ←相互に模倣可能⇒ Turing機械



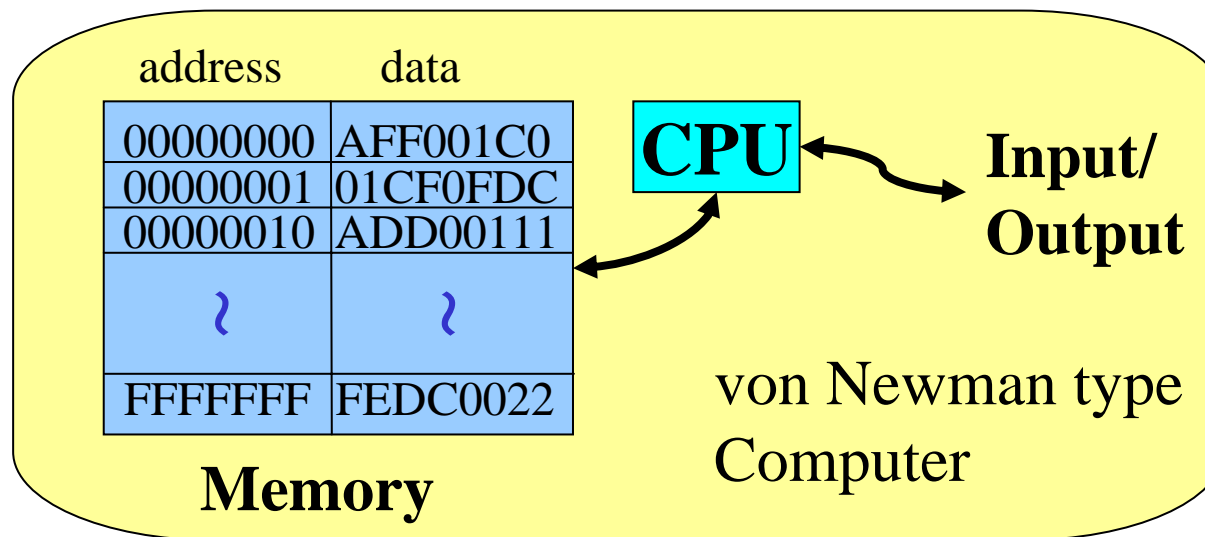
★ Turing機械
そのもので
Turing機械を
模倣すること
もできる。

万能Turing機械: 与えられた TM のコードを実行(模倣)するTM

8.6. Turing machine and Computer

- Turing machine has **universality**, and has the **same computation power** as usual computer

'Usual computer' \Leftarrow Simulate each other \Rightarrow Turing Machine



★ Turing Machine can simulate another Turing Machine.

Universal Turing Machine: TM that simulates given code of TM.

今後の予定(Schedule)

	授業(Lecture)	Office Hour
May 24, 26	休講(Canceled)	---
May 31 (Wed)	計算の可能性 (Unsolvability) 授業アンケート (questionnaire)	レポート4,5の解答と解説 (Answer and Comments to report (4) and (5))
June 2(Fri)	テスト(Final Exam)	---

範囲: 1章~7章

Scope: Chapter 1~Chapter 7

おまけ情報

- これまでの授業のビデオ (Videos taken in this class):

<http://jenzabar.jaist.ac.jp/>

- 田町キャンパス用のスタジオ撮影ビデオを試験的公開
(Temporal videos for Tamachi-campus taken in studio):

<http://e-tamachi.jaist.ac.jp/tamachi06/lectures06/i1113/index.html>

- 成績に関する問い合わせは uehara@jaist.ac.jp まで
Feel free to ask uehara@jaist.ac.jp about your records.

- 上記のリンクは授業の Web ページからリンクをはってあります。
(You can click above links from my web page for this class.)