授業アンケート実施「教員独自の設問」:

(1) 数学科の他の情報系科目との連携は
適切だったと思いますか
(独立し過ぎ ← 適切 → 重複し過ぎ)

(2) 数学科の教職課程 (教科「数学」) の
コンピュータ区分の科目として
適切な内容だったと思いますか
(専門的過ぎ ← 適切 → 概論的過ぎ)

(3) 数学科の 3 年配当の科目として
適切な内容だったと思いますか
(専門的過ぎ ← 適切 → 概論的過ぎ)

期末試験のお知らせ

7月27日(月) 15:15 〜 16:15 (60分試験)

9-255 教室 (ここじゃない)

- 最終回 (7/20) の講義内容まで
- 学生証必携

レポート提出について

期日: 8月7日(金)20時頃まで

内容:

配布プリントのレポート問題の例のような内容及び授業に関連する内容で、
授業内容の理解または発展的な取組みをアピールできるようなもの

提出方法:

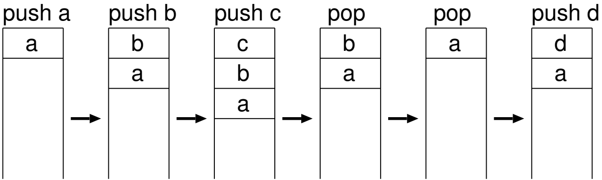
- **授業時に手渡し**
- **4-574室扉のレポートポスト**
- **電子メイル**

代表的な計算モデル

- 有限オートマトン
- プッシュダウンオートマトン
- チューリングマシン

プッシュダウンオートマトン

(非決定性) 有限オートマトンに
プッシュダウンスタックを取り付けたもの



無限 (非有界) の情報を保持できるが、
読み書きは先頭だけ

$\cdots$ **LIFO (Last In First Out)**

定理:

L : **文脈自由言語**

$$\Updownarrow$$

L **が或るプッシュダウンオートマトンで認識される**

プッシュダウンオートマトンでは認識できない言語の例

同じ文字列 2 回の繰返しから成る文字列全体

$$A = \{ww | w \in \Sigma^*\}$$

入力を読み直せないのが弱点

$\longrightarrow$ **より強力な計算モデルが必要**

一つの方法としては、

入力を覚えておくために
プッシュダウンスタックをもう一つ
使えることにする

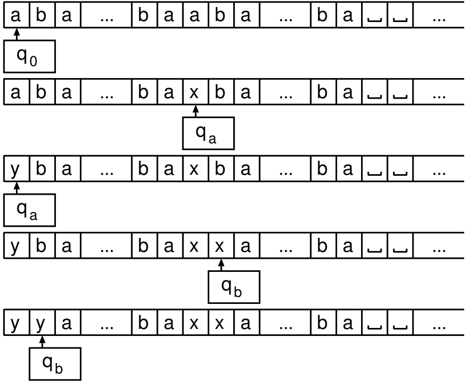
実際これで真により強い計算モデルが得られる

しかし、通常はこれと同等な
次のような計算モデルを考える
· · · チューリングマシン

チューリングマシン

- 有限個の内部状態を持つ
- 入力データはテープ上に一区画一文字づつ書き込まれて与えられる
- データを読み書きするヘッドがテープ上を動く
- 遷移関数は次の形:
  内部状態とヘッドが今いる場所の文字とによって、その場所の文字を書き換え、次の内部状態に移り、ヘッドを左か右かに動かす
- 受理状態または拒否状態に達したら停止するが、停止しないこともある

(非決定性) チューリングマシンによる言語 $A = \{ww | w \in \Sigma^*\}$ の認識



チューリングマシン T が言語 A を**認識**する

$$\Updownarrow$$

$$A = \left\{ w \in \Sigma^* \;\middle|\; \begin{array}{l} \text{入力 } w \text{ に対し、} \\ \text{受理状態で停止する} \\ \hfill \text{遷移が存在} \end{array} \right\}$$

$$\Updownarrow$$

$$w \in A \Longleftrightarrow \begin{array}{l} \text{入力 } w \text{ に対し、} \\ \text{受理状態で停止する遷移が存在} \end{array}$$

チューリングマシン T が言語 A を判定する

$$\Updownarrow$$

T は A を認識し、
かつ、全ての入力に対し必ず停止する

$$\Updownarrow$$

$w \in A \iff$ **入力 w に対し、受理状態で停止する遷移が存在**

かつ

$w \notin A \iff$ **入力 w に対し、拒否状態で停止する遷移が存在**

Church-Turing の提唱

「全てのアルゴリズム (計算手順) は、
チューリングマシンで実装できる」

(アルゴリズムと呼べるのは
チューリングマシンで実装できるものだけ)

… 「アルゴリズム」の定式化

Church-Turing の提唱

「全てのアルゴリズム (計算手順) は、
チューリングマシンで実装できる」

(アルゴリズムと呼べるのは
チューリングマシンで実装できるものだけ)

· · · 「アルゴリズム」の定式化

Church-Turing の提唱

「全てのアルゴリズム (計算手順) は、
チューリングマシンで実装できる」

(アルゴリズムと呼べるのは
チューリングマシンで実装できるものだけ)

$\cdots$ **「アルゴリズム」の定式化**

何故「チューリングマシン」なのか?

- **およそ計算機で実行したいことは模倣可能 (無限のメモリにランダムアクセスできる計算機モデル)**
- **多少モデルを変更しても強さが同じ (モデルの頑強性)**
  - ⋆ テープが両方に無限に伸びているか
  - ⋆ ヘッドが動かなくても良いか
  - ⋆ 複数テープチューリングマシン
  - ⋆ 決定性／非決定性　　　など など

プログラム内蔵方式 (von Neumann 型)
$\cdots$ プログラムもデータとして保持
$\longrightarrow$ 一つの機械で様々な計算を柔軟に実現

同様の働きをする
チューリングマシン U が構成できる

$\cdots$ 万能チューリングマシン
(universal Turing machine)

プログラム内蔵方式 (von Neumann 型)
$\cdots$ **プログラムもデータとして保持**
$\longrightarrow$ **一つの機械で様々な計算を柔軟に実現**

同様の働きをする
チューリングマシン U が構成できる

$\cdots$ **万能チューリングマシン**
(universal Turing machine)

プログラム内蔵方式 (von Neumann 型)
$\cdots$ **プログラムもデータとして保持**
$\longrightarrow$ **一つの機械で様々な計算を柔軟に実現**

同様の働きをする
チューリングマシン U が構成できる

$\cdots$ **万能チューリングマシン**
(universal Turing machine)

万能チューリングマシン

全てのチューリングマシンの動作を模倣する

入力: $(\langle M\rangle, w)$

- $\langle M\rangle$ **: 機械** M **の符号化**
  (プログラムに相当)
- w **:** M **に与える入力データ**

出力: 機械 M **が入力** w **を受理するかどうか**

定理

言語

$$A_{\mathrm{TM}} = \left\{ (\langle M \rangle, w) \;\middle|\; \begin{array}{l} \langle M \rangle : \textbf{TM } M \textbf{ の符号化} \\ M \textbf{ が入力 } w \textbf{ を受理} \end{array} \right\}$$

は認識可能だが、判定可能ではない。

証明は一種の対角線論法による。
(Russell のパラドックス風)

定理

言語

$$A_{\mathtt{TM}} = \left\{ (\langle M \rangle, w) \;\middle|\; \begin{array}{l} \langle M \rangle : \textbf{TM } M \textbf{ の符号化} \\ M \textbf{ が入力 } w \textbf{ を受理} \end{array} \right\}$$

は認識可能だが、判定可能ではない。

証明は一種の対角線論法による。
(Russell のパラドックス風)

参考: Russell のパラドックス

$X := \{A | A \notin A\}$ とせよ

$X \in X$ であるか？

- $X \in X$ と仮定すると、定義より $X \notin X$
- $X \notin X$ と仮定すると、定義より $X \in X$

$\longrightarrow$ どちらにしても矛盾!!

参考: Russell のパラドックス

$X := \{A | A \notin A\}$ とせよ

$X \in X$ であるか？

- $X \in X$ と仮定すると、定義より $X \notin X$
- $X \notin X$ と仮定すると、定義より $X \in X$

$\longrightarrow$ どちらにしても矛盾!!

参考: Russell のパラドックス

$X := \{A | A \notin A\}$ とせよ

$X \in X$ であるか？

- $X \in X$ と仮定すると、定義より $X \notin X$
- $X \notin X$ と仮定すると、定義より $X \in X$

$\longrightarrow$ どちらにしても矛盾!!

参考: Russell のパラドックス

$X := \{A | A \notin A\}$ とせよ

$X \in X$ であるか？

- $X \in X$ と仮定すると、定義より $X \notin X$
- $X \notin X$ と仮定すると、定義より $X \in X$

$\longrightarrow$ どちらにしても矛盾!!

参考: Russell のパラドックス

$X := \{A | A \notin A\}$ とせよ

$X \in X$ であるか？

- $X \in X$ と仮定すると、定義より $X \notin X$
- $X \notin X$ と仮定すると、定義より $X \in X$

$\longrightarrow$ どちらにしても**矛盾**!!

$A_{\mathtt{TM}}$ の判定不可能性

$A_{\mathtt{TM}}$ を判定する TM U があったとする。

入力 $\langle M \rangle$ に対し、

- **M が $\langle M \rangle$ を受理するなら拒否**
- **M が $\langle M \rangle$ を拒否するなら受理**

となる TM D が (U を使って) 作れる。

これに、入力 $\langle D \rangle$ を喰わせよ。

$A_{\mathtt{TM}}$ の判定不可能性

$A_{\mathtt{TM}}$ を判定する TM U があったとする。

入力 $\langle M \rangle$ に対し、

- M が $\langle M \rangle$ を受理するなら拒否
- M が $\langle M \rangle$ を拒否するなら受理

となる TM D が (U を使って) 作れる。

これに、入力 $\langle D \rangle$ を喰わせよ。

$A_{\mathtt{TM}}$ の判定不可能性

$A_{\mathtt{TM}}$ を判定する TM U があったとする。

入力 $\langle M \rangle$ に対し、

- M が $\langle M \rangle$ を受理するなら拒否
- M が $\langle M \rangle$ を拒否するなら受理

となる TM D が (U を使って) 作れる。

これに、入力 $\langle D \rangle$ を喰わせよ。

さて、本講義最後の話題は、

計算量

について

問題の難しさを如何に計るか？

さて、本講義最後の話題は、

計算量

について

問題の難しさを如何に計るか？