

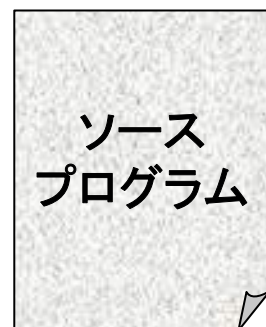
# コンピュータ概論 第8回

---

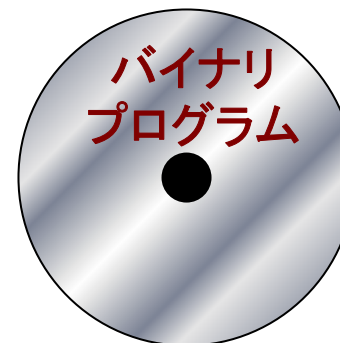
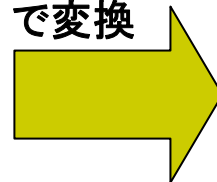
アルゴリズムとは

# プログラムとは

- コンピュータを動作させる命令
  - 処理装置の動作命令の集まり
  - 動作手順も規定 (program = 計画表)
  - 処理装置は命令を逐次 (1つ1つ順番に) 実行
- プログラミング言語
  - ソースプログラムを作成するための人工言語
  - アルゴリズムを記述しやすい



言語処理  
プログラム  
で変換



# アルゴリズムとは

---

## □ 定義(大辞林より引用)

### ■ アルゴリズム [algorithm]

[アラビアの数学者アル＝フワリズミの名にちなむ]

(1) もとは算用数字を用いた筆算のこと。

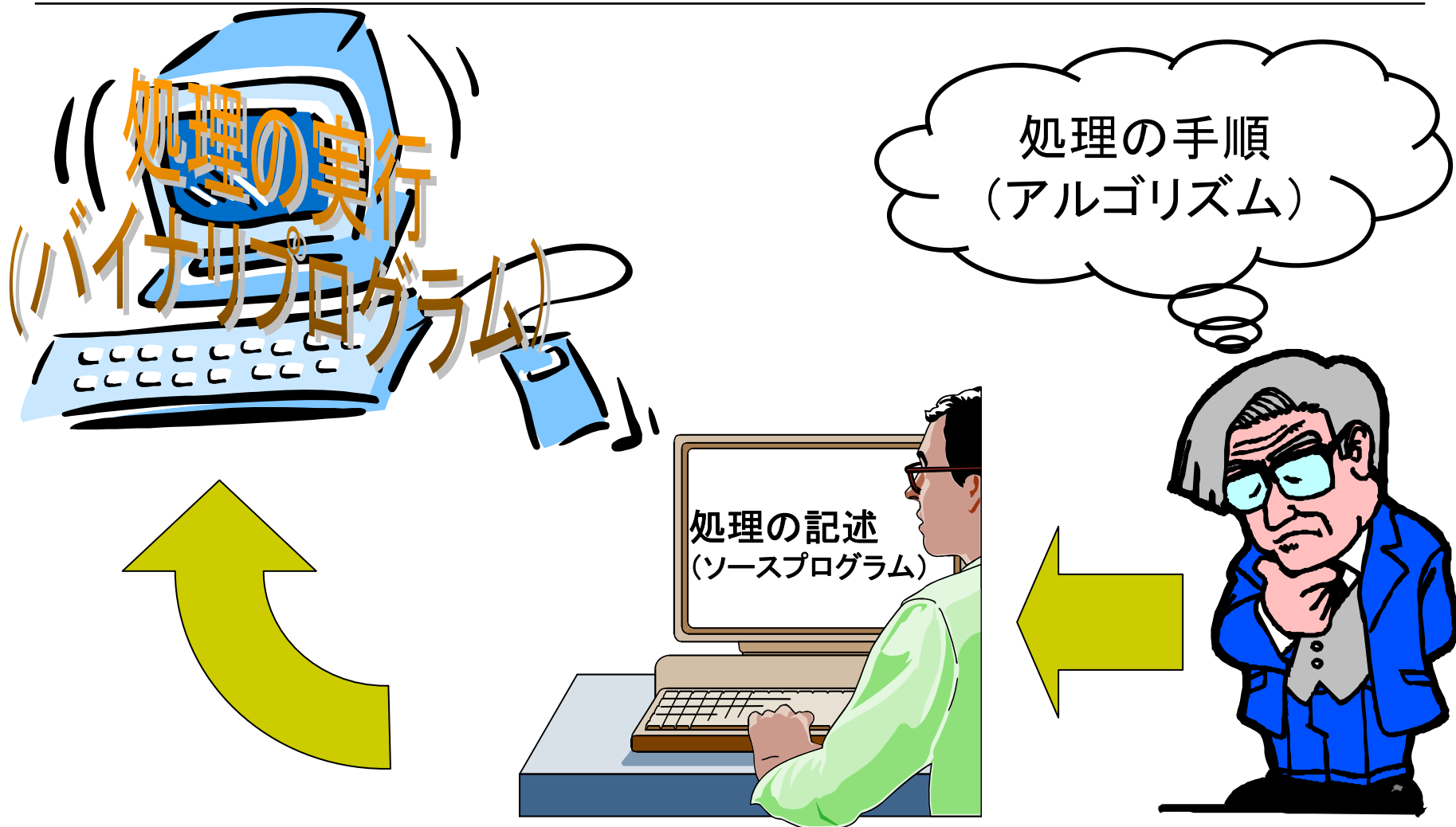
(2) 計算や問題を解決するための手順、方式。特にコンピューターのプログラムに適用可能な手続きをいうことが多い。

## □ 本講義での定義

■ ある目的を達成するための手続きおよびその手順

■ コンピュータの動作(プログラム)の論理的な内容

# プログラムとアルゴリズム



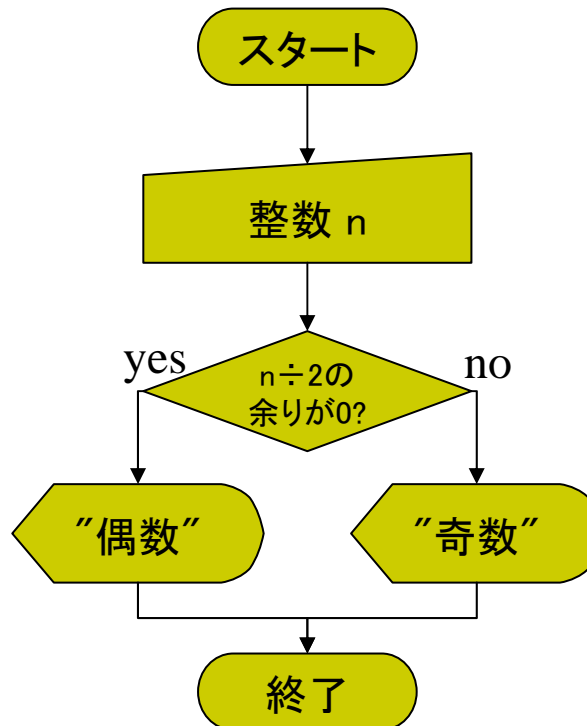
# アルゴリズムの表現方法

---

- 文字(文章)による表現
  - 自然言語での表現
    - 文章による表現
    - 箇条書きによる表現
  - 人工言語による表現
    - プログラミング言語による記述
    - 仕様記述言語による記述
- 図による表現
  - フローチャート(流れ図)による記述
  - その他の図表(NSチャートなど)による記述

# フローチャート

- アルゴリズムの流れが分かるようにした図
- 図中で使える記号には決まりがある



# フローチャートの構成要素

---

端子(始まりと終わり)



流れ(推移)



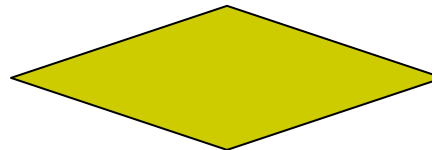
合流



処理(手続き)



条件判断



手入力





# アルゴリズムの例

---

- 連立方程式の解法
- 10進数を2進数に変換する方法
- ATMでお金を引き出す操作方法  
など
  
- 目的を達成するための手順を伴う手続き  
であれば、何でもアルゴリズムと呼べる



## 例題 身近なものでアルゴリズムを考える

---

- 手順のある仕事
- 曖昧な部分が少ない事
- わかりやすい事



ガツパ焼きそば



# カップ焼きそばとは？

---

- カップラーメンの焼きそば版
  - 熱湯があれば調理道具無しで作れる
- 特徴
  - 単にお湯を入れて出来上がる  
カップラーメンよりは作り方が複雑
  - 同種のものにカップパスタなどもある
  - 作り方の手順を間違えると・・・

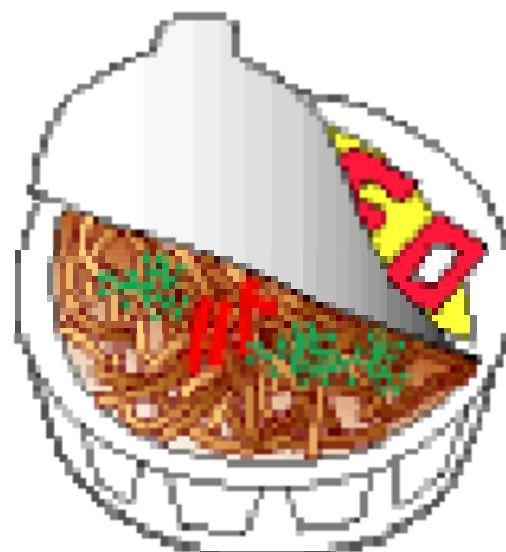
# カップ焼きそばの内訳

---

- 発泡容器
  - カップラーメンと同様
- 湯切り穴のあるふた
- 乾燥麺
- かやく
  - 袋入りの乾燥された具（肉やキャベツなど）
- 薬味
  - 袋入りの青海苔や紅しょうが
- ソース
  - 液体で袋入り
- 割り箸

# 練習「カップ焼きそば調理 アルゴリズムを考えよ」

- カップ焼きそばは容器のふたを少し開け、  
中からかやくと薬味、ソースを取り出してある。  
(出し忘れてお湯を注ぐととても悲しい)
- 熱湯はポットに準備されている。



# 手続き一覧

---

- ① 湯を注ぐ
- ② 湯を捨てる
- ③ かやく(乾燥具)を容器に入れる
- ④ ソースを入れる
- ⑤ 薬味をいれる
- ⑥ ふたをする(戻す)
- ⑦ ふたを完全にはがす
- ⑧ かき混ぜる
- ⑨ 茹であがり時間(3~5分)待つ

# 模範解答

---

- ③ かやく(乾燥具)を容器に入れる
- ① 湯を注ぐ
- ⑥ ふたをする(戻す)
- ⑨ 茹であがり時間(3~5分)待つ
- ② 湯を捨てる
- ⑦ ふたを完全にはがす
- ④ ソースを入れる
- ⑧ かき混ぜる
- ⑤ 薬味をいれる

※④⑧⑤の順序は入れ替えても問題なし



# ありがちな誤ったアルゴリズム

---

- ②お湯を捨てる前に④ソースを入れる
  - 薄味で味足らず(はじめて作る人に多い)
- ②お湯を捨てる前に⑦ふたを完全にはがす
  - お湯を捨てるのが大変(やけどに注意!)
- ②お湯を捨てる前に⑤薬味を入れる
  - 薬味が流れ出てしまう(青のりが!!)
- ②お湯を捨てた後で③かやくを入れる
  - 具が硬くて美味しくない(麺の下に埋めてみる)

アルゴリズムを誤ると目的が達成できない



# 正しいアルゴリズム

---

## □ アルゴリズムの定義

ある目的を達成するための手続きおよびその手順

## □ 正しいアルゴリズムの数

- 目的を達成するための手続きおよびその手順は何通りでも存在する

- 正しく目的達成できれば、それらは全て正しいアルゴリズムといえる





# 制御構造

---

## アルゴリズムの流れを制御する構造

### □ 制御構造

#### ■ 順次構造

- 目的地まで一直線

#### ■ 分岐構造

- 分かれ道を選んで

#### ■ 繰返構造

- 必要に応じて後戻りしながら

# 条件判断

---

- 制御構造に不可欠な処理
  - 真か偽かを判定する
    - 分岐構造      処理を2つの流れに分ける
    - 繰返構造      処理を繰り返す
  - 条件判断の例
    - 変数xは正の数か？
    - 変数yは大文字か？

# 例題 身近なもので アルゴリズムを考える

---

- 手順のある仕事
- 曖昧な部分が少ない事
- わかりやすい事



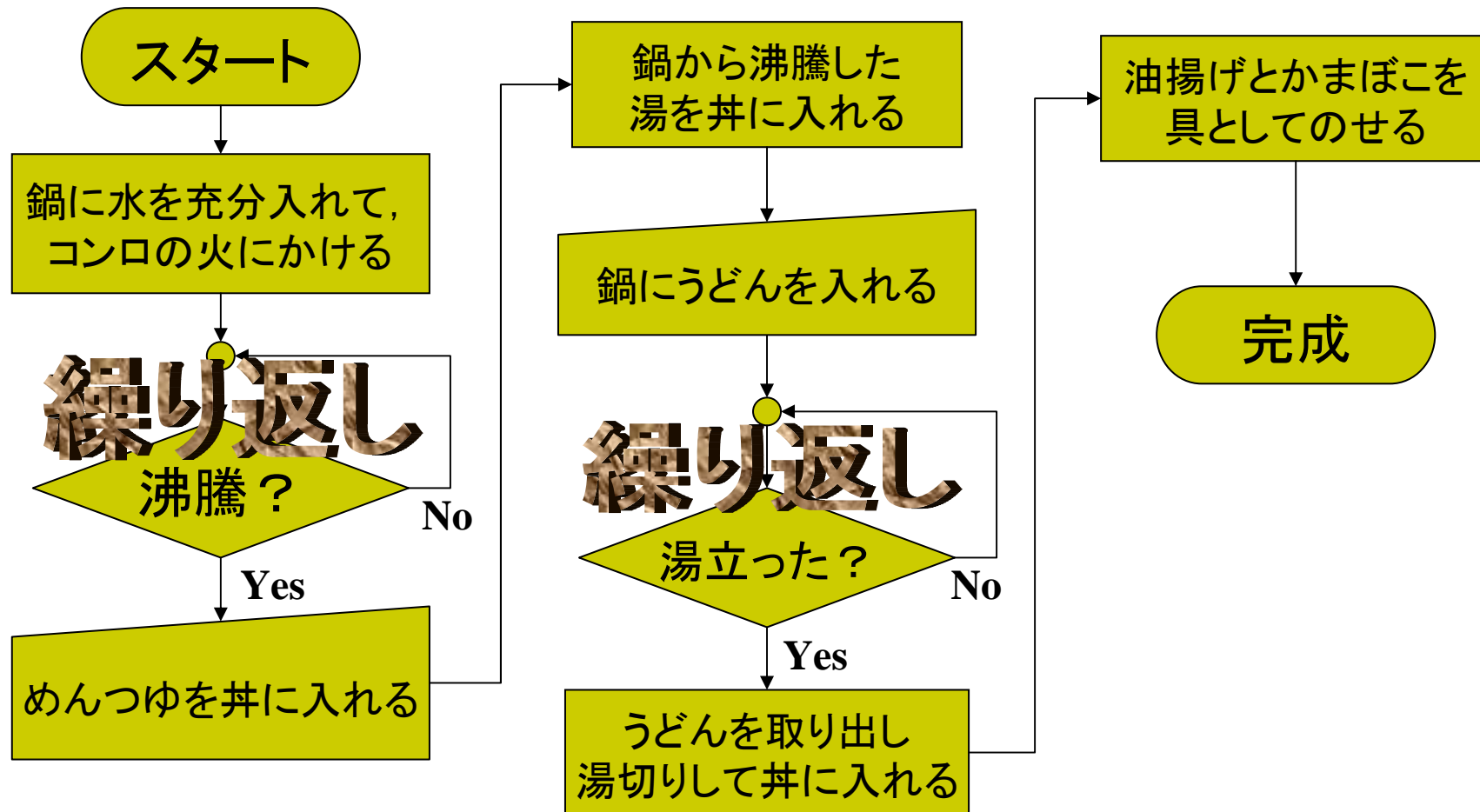


# きつねうどん調理の流れ(概要)

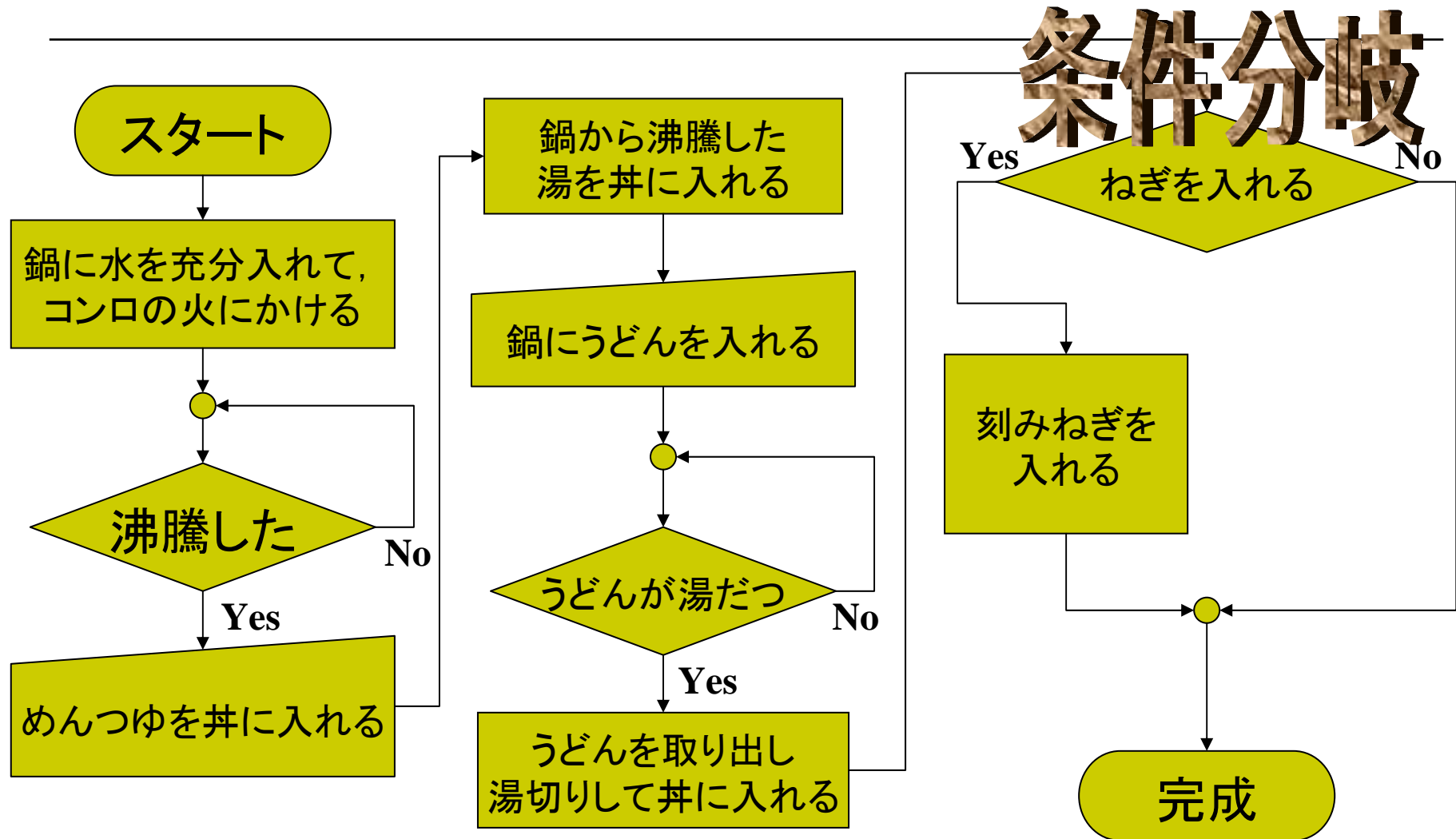
---


- 鍋に水を充分入れて, コンロの火にかける
- 沸騰するまで待つ
- めんつゆを適量どんぶりに入れる
- 沸騰した湯をどんぶりに入れる
- 湯の残った鍋を再び火にかけるうどんを入れる
- うどんが湯だつまで待つ
- うどんを取り出し, どんぶりに入れる
- 油揚げとかまぼこをどんぶりに入れる

# きつねうどんフローチャート



# すうどんフローチャート





# 条件判断

---

## □ 条件判断の種類

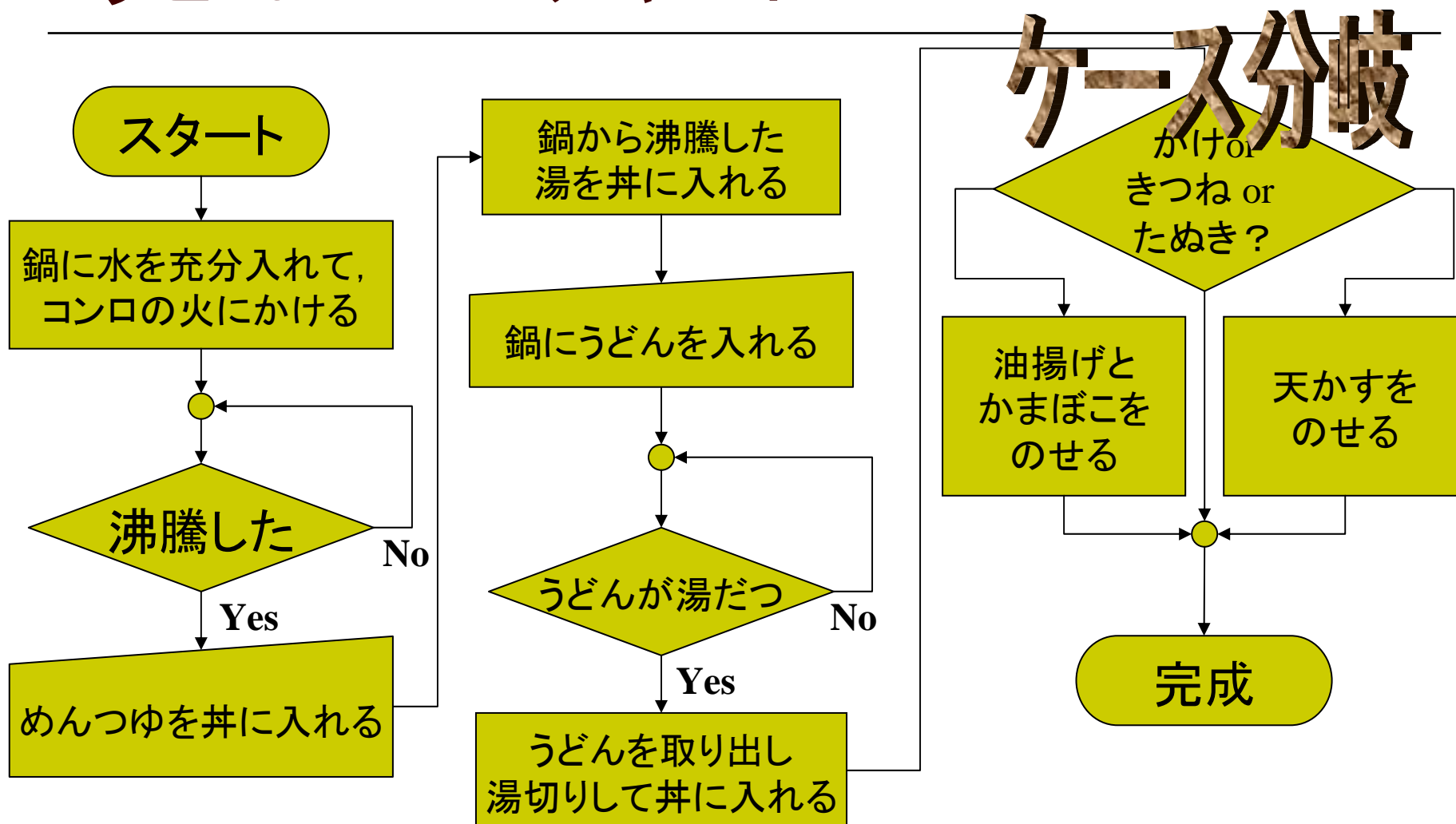
### ■ Yes or No

- 条件分岐      処理を2つの流れに分ける
- 繰り返し      処理を繰り返す

### ■ 場合分け

- ケース分岐    複数の流れ（ケース毎）に分ける

# うどんフローチャート







# より複雑なアルゴリズム

---

- サブルーチン(サブアルゴリズム)
  - アルゴリズムの一部を、1つの手続きとみなしてまとめることで全体を簡略化
  - サブルーチンも1つのアルゴリズムといえる
  - まとめかたは自由
    - さまざまな手法がある
    - まとめ方が下手だとアルゴリズムは複雑になる

# サブルーチン

- 手続き:「鍋に水を充分入れて, コンロの火にかける」  
これは1つの処理か?

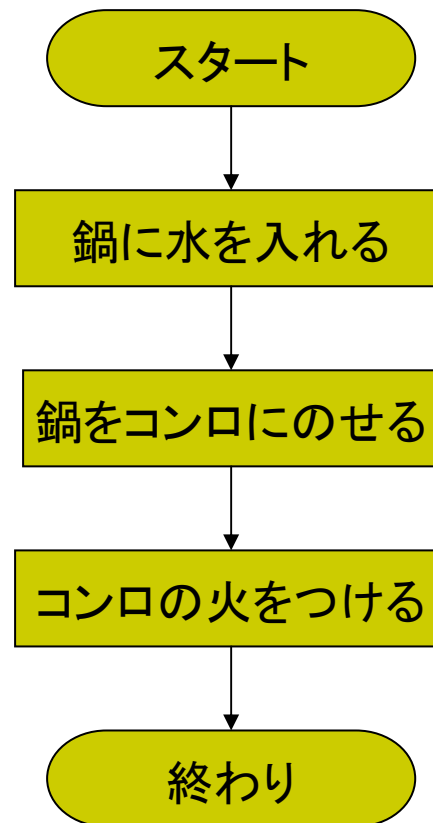
処理(手続き)

鍋に水を充分入れて,  
コンロの火にかける

定義済み処理(手続き)

鍋に水を充分入れて,  
コンロの火にかける

Cの関数はこれに相当  
(詳しくは12回で)





# おさらい

---

- プログラムとアルゴリズムの関係
- 実世界におけるアルゴリズム
- フローチャート
- 複雑なアルゴリズム
  - 制御構造
  - 条件判断
  - サブルーチン

# プログラム課題

---

- キーボードからアルファベットの小文字を1文字入力して、大文字を表示するプログラムを作成せよ
  - ファイル名 : cap.c
  - 結果を表示する時には、入力した文字と結果を、「入力された文字～の大文字は～です」のように表示すること。
  - ヒント 大文字の文字コードは小文字の文字コードより32小さい