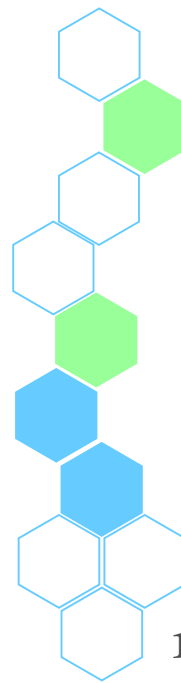


情報処理技法 (Javaプログラミング)2

第3回 アルゴリズム(1)

人間科学科コミュニケーション専攻
白銀 純子



第3回の内容



◆ アルゴリズム



前回の出席課題の解答

- ◆ 下記のプログラムを実行したとき、確実に標準出力に出力されるメッセージは何かを答えなさい。

```
try {
    String line[] = new String[5];
    int i;

    FileReader fr = new FileReader("sample.txt");
    BufferedReader br = new BufferedReader(fr);

    for(i = 0; i < 100; i = i + 1) {
        line[i] = br.readLine();
    }
    br.close();
    fr.close();
}
catch (ArrayIndexOutOfBoundsException e) {
    System.out.println("指定された範囲外の添え字を使おうとしています。");
}
catch (NumberFormatException e) {
    System.out.println("数値に変換できない文字列を数値に変換しようとしています。");
}
catch (FileNotFoundException e) {
    System.out.println("指定されたファイルが存在しません。");
}
catch (IOException e) {
    System.out.println("指定されたファイルを読み込むことができません。");
}
```

確実に標準出力に
出力されるメッセージ

アルゴリズム



アルゴリズムとは



- ◆ アルゴリズム: ある問題を解決するときに必要な処理手順
 - ◆ プログラムを書くときには、必ずアルゴリズムを考える必要
 - ◆ プログラム: アルゴリズムをプログラミング言語を使って記述したもの

Ex. 焼きそば作りのアルゴリズム:

1. キャベツや肉などの具をフライパンで炒めなさい。
2. フライパンからいったん具を取り出さなさい。
3. めんをフライパンで炒めなさい。
4. 具をめんの入っているフライパンに戻しなさい。
5. 焼きそばソースを加えてさらに炒めなさい。

アルゴリズムの表現方法

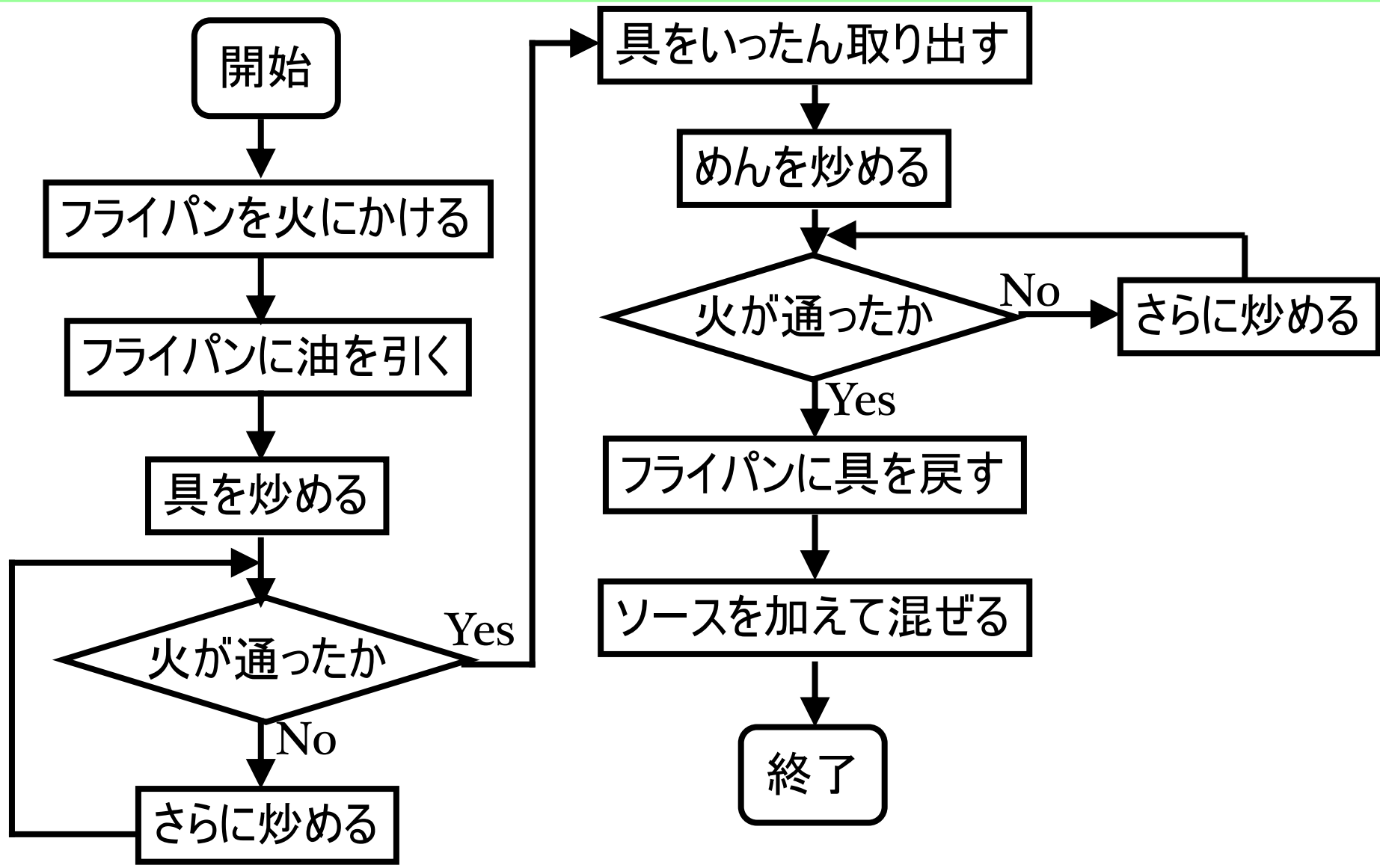


- ◆ 文章で書く
 - ◆ 箇条書きで書くことも多い
- ◆ 図で書く
- ◆ プログラムで書く
 - ◆ プログラムが最終段階



プログラムで行う処理手順がわからなければ、まず文章で書き出して、それを詳細にしていく

焼きそば作成(図)



焼きそば作成(プログラム1)



焼きそばのアルゴリズムをプログラムの的に表現

```
public static void main(String[] args) {  
    // 開始  
    フライパンを火にかける;  
    フライパンに油を引く;  
    具を炒める;  
    while (火が通っていない) {  
        // Yes(「火が通ったか?」に対して「No」)  
        さらに炒める  
    }  
    // 「火が通ったか?」に対して「Yes」  
    具をいったん取り出す;  
    めんを炒める;  
}
```

焼きそば作成(プログラム2)



焼きそばのアルゴリズムをプログラムの表現(続き)

```
while (火が通っていない) {  
    // Yes(「火が通ったか?」に対して「No」)  
    さらに炒める;  
}  
// 「火が通ったか?」に対して「Yes」  
フライパンに具を戻す;  
ソースを加えて混ぜる;  
// 終了  
}
```

最大公約数のアルゴリズム



素因数分解



1. 1つ目の数を素因数分解する
2. 2つ目の数を素因数分解する
3. それぞれの素因数の共通項を掛け合わせる

Ex. 136と24の最大公約数

$$136 = 2 \times 2 \times 2 \times 17$$

$$24 = 2 \times 2 \times 2 \times 3$$

➡ 共通項: $2 \times 2 \times 2 = 8$

➡ 最大公約数: 8

最大公約数



- ◆ 2つの数の最大公約数を求めるアルゴリズムは2通り
 - ◆ 素因数分解で求める
 - ◆ ユークリッドの互助法で求める

ユークリッドの互助法



1. 2つの数のうち、大きい方を小さい方で割る
2. 1. の余りが0でない場合、その余りと小さい方の数を改めて2つの数とし、1. へ戻る
3. 1. の余りが0の場合、割った数が最大公約数となる

Ex. 136と24の最大公約数

$$136 \div 24 = 5 \dots 16$$



$$24 \div 16 = 1 \dots 8$$

$$16 \div 8 = 2 \dots 0$$



最大公約数: 8

筆算のアルゴリズム



自然数の加算(1)



◆ 2つの自然数の和を求める = 筆算

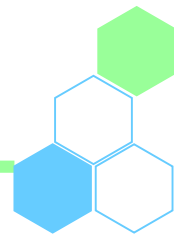
- ◆ n 桁目の数の和を求め、10以上になった場合には $n+1$ 桁目に1を加える

n が1から自然数の桁数まで繰り返す

◆ 準備

- ◆ 2つの自然数は同じ桁数とする
 - ◆ 50と200の場合、050と200として考える
- ◆ 加えた結果を入れる変数(配列が便利)を用意する
- ◆ 繰り上がりを入れる変数を用意し、0を代入しておく

自然数の加算(2)



1. 1の位から最上位の位まで、a. ～d. を繰り返す
 - a. その位の2つの数の和を求める
 - b. 下位からの繰り上がりがあればそれに加える
 - c. その和の1桁目を、求める和のその位の結果とする
 - d. その和の2桁目を、次の桁への繰り上がりとする
2. 最上位からの繰り上がりがあった場合は、求める和のその次(上位)の位の数とする

ATMのアルゴリズム



ATM(1)



■ 人間から見える手順

1. 「引き出し」ボタンを押す
2. キャッシュカードを入れる
3. 暗証番号を入力する
4. 引き出し金額を入力する
5. 現金を取り出す



◆ ATM側での処理手順

1. 初期画面を表示してボタンが押されるのを待つ
2. 押されたボタンによってサービスの種類を判別する
 - a. 「引き出し」ボタンの場合、3. へ進む
 - b. 「預け入れ」ボタンの場合、4. へ進む

.....
3. サービスの種類が「引き出し」の処理
4. サービスの種類が「預け入れ」の処理
.....
5. 「ありがとうございました」の画面を表示して、1. へ戻る



◆ 「引き出し」の処理手順

- a. カードを入れてくださいと表示する
- b. 適切なカードかどうかを調べ、キャッシュカードではないカードであれば一.へ進む
 - 一. 入れられたカードを排出し、「キャッシュカードを入れなおしてください。」と表示する
 - 二. b.へ戻る
- c. 暗証番号を調べ、番号が間違っていたら一.へ進む
 - 一. 「暗証番号を入力しなおしてください。」と表示する
 - 二. c.へ戻る
- d. 「引き出す金額を入力してください。」と表示し、引き出し金額を入力してもらう
- e. 引き出し金額を調べる



◆ 「引き出し」の処理手順の続き

- f. 引き出し金額が残高よりも多ければ、一. へ進む
 - 一. 「残高を超えています。引き出し金額を入力しなおしてください。」と表示する
 - 二. f. へ戻る
- g. 出金処理を行い、預金残高を更新する
- h. 預金残高を表示し、5. へ進む

探索(1)



◆ たくさんのデータから目的のデータを見つけること

Ex. 高校の生徒の得点を管理するプログラム

- 出席番号5番の生徒の英語の点数を知りたい
→ 高校の生徒のデータから、出席番号が「5」というものを探す
- 「東京子」という生徒の国語の点数を知りたい
→ 高校の生徒のデータから、名前が「東京子」というものを探す

探索(2)



◆ 探索のアルゴリズム

- ◆ 逐次探索
- ◆ 2分探索
- ◆ 自己組織化探索
- ◆ 2次元探索
- ◆ 補間探索
- ◆ ハッシュ法



逐次探索

- ◆ データを前から順番に比較して探していく方法
 1. データが配列 $a[0] \sim a[N]$ に入っている
 2. 「 $a[i] = x$ 」(x は探したいデータ)となる「 i 」を探す
 - ◆ 「 i 」は $0 \sim N$ の整数

- ◆ 欠点: データが配列の後ろのほうにある場合に見つけるまでに時間がかかる



やってみよう!(1)

◆ ユーグリッドの互助法のプログラム

- ◆ 2つの数を標準入力で

◆ ATMの「引き出し」と「預け入れ」の機能のプログラム

- ◆ 預金残高と暗証番号は、あらかじめプログラム中に書いておく
- ◆ カードに関する処理はなしでOK
- ◆ 出金処理と入金処理は、標準入力で処理
 - ◆ 出金: 「xx円出金し、yy円残っています。」と表示
 - ◆ 入金: 入金する金額を標準入力で入力

やってみよう!(2)



- ◆ 「アルゴリズム」という教材に挑戦してみよう

- ◆ <http://home.jeita.or.jp/is/highschool/algo/index.html>

- ◆ アルゴリズムの考え方を身につけるのに良い教材