

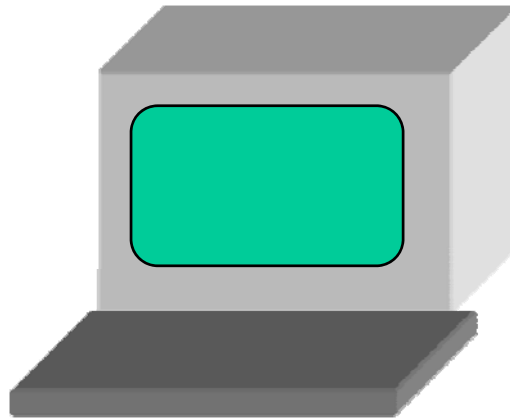
第1回プログラミング入門

(教科書1～3章)



プログラミング演習の目的

コンピュータを用いた問題解決ができるようになる。
そのために、プログラムの作成能力を身に付ける。



今回の目標

- 現実の問題をプログラムにするまでの概要を理解する。
- 課題の提出法を習得する。

☆演習室から課題を提出する。

コンピュータの2つの側面

ソフトウェアとハードウェア

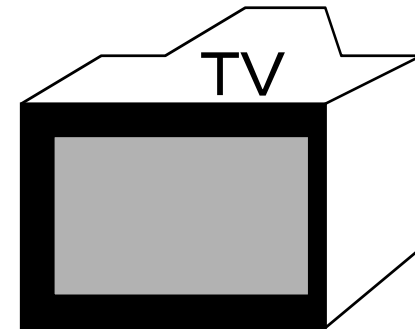
ソフトウェア

プログラム

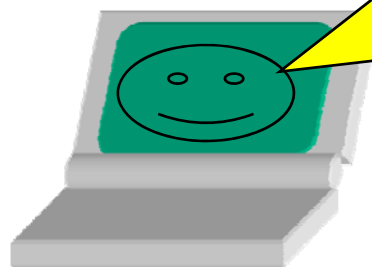
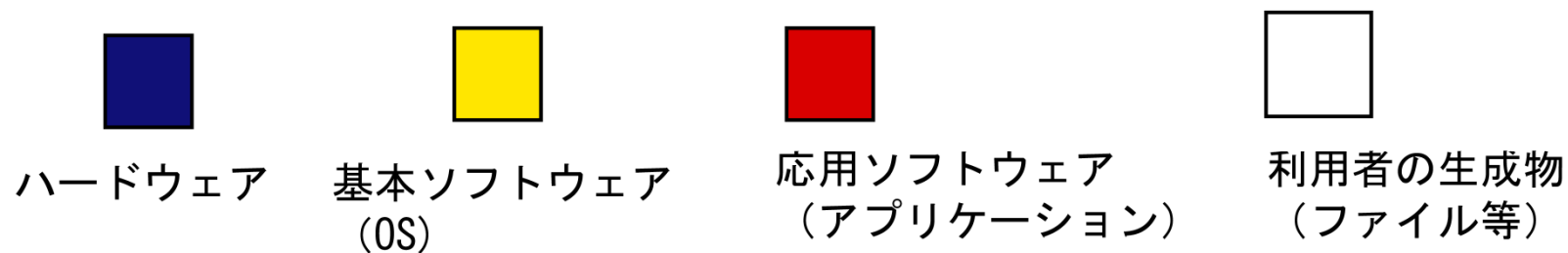
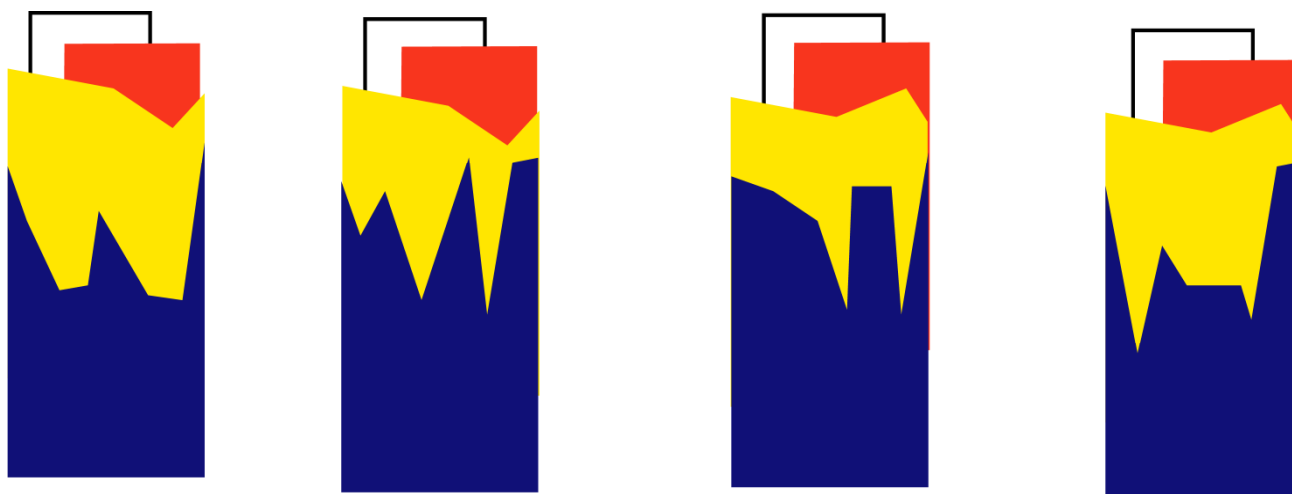
番組

思考、記憶

ハードウェア

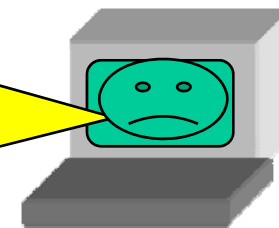


ハードウェアとソフトウェアの階層構造

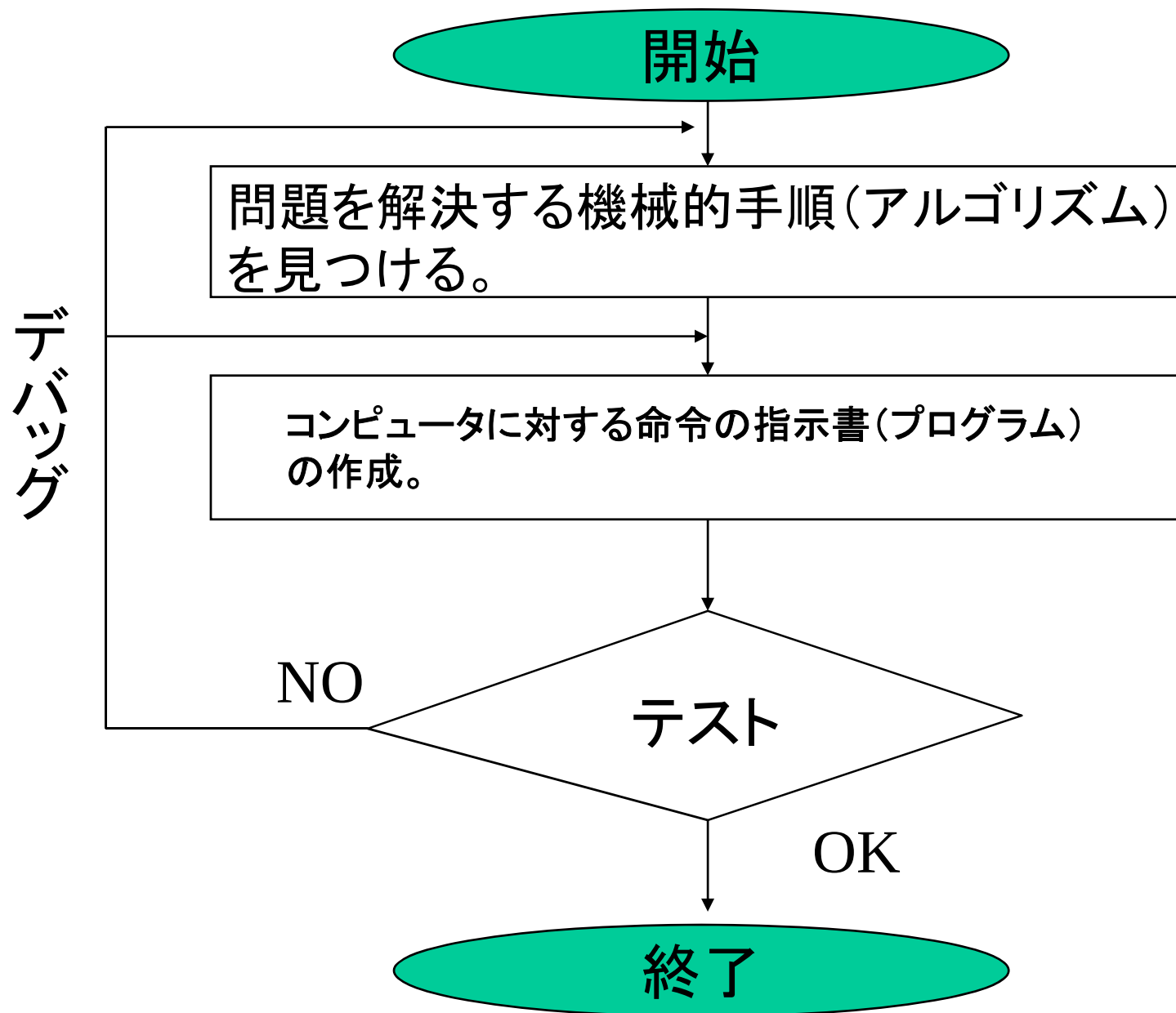


自分用の
0と1の並びしか
わからないよ。

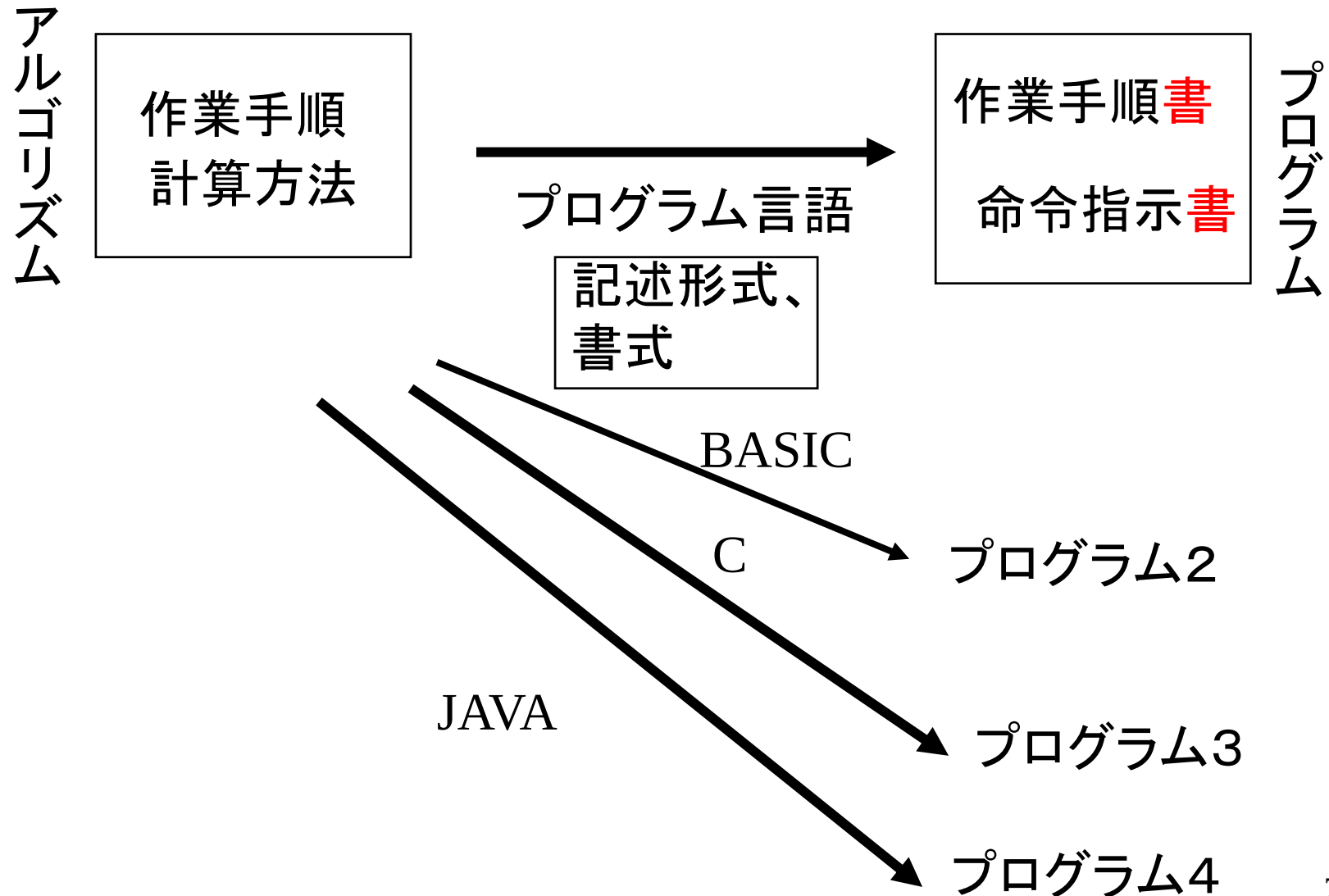
他のコンピュータ
は、別の0と1の
並びを用いてる
かも。



問題解決のためのソフトウェアの開発



アルゴリズムとプログラム

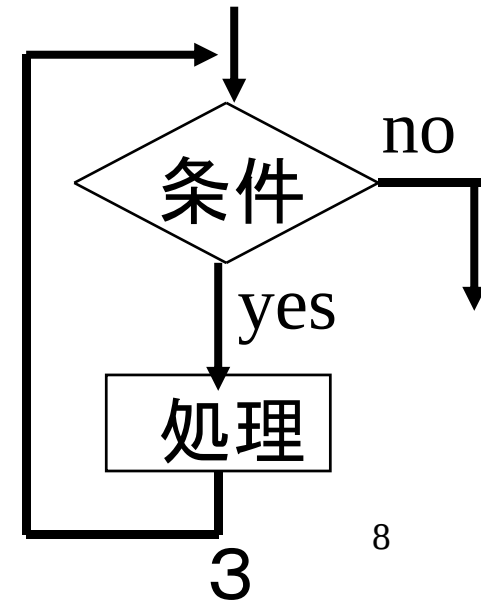
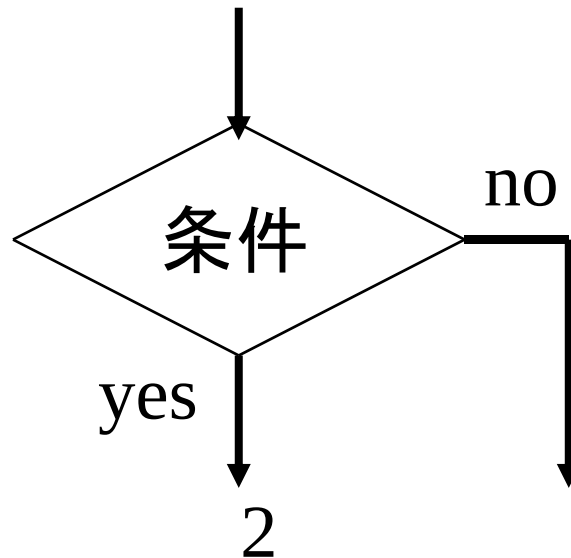
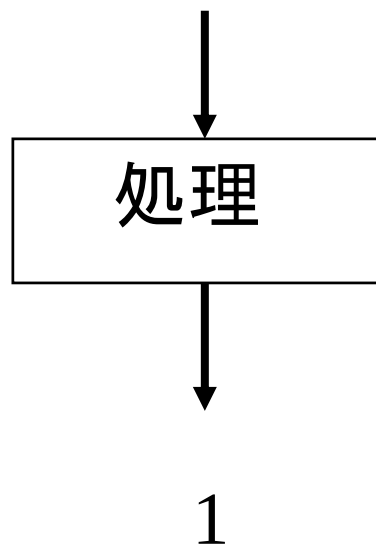


アルゴリズムの記述の仕方

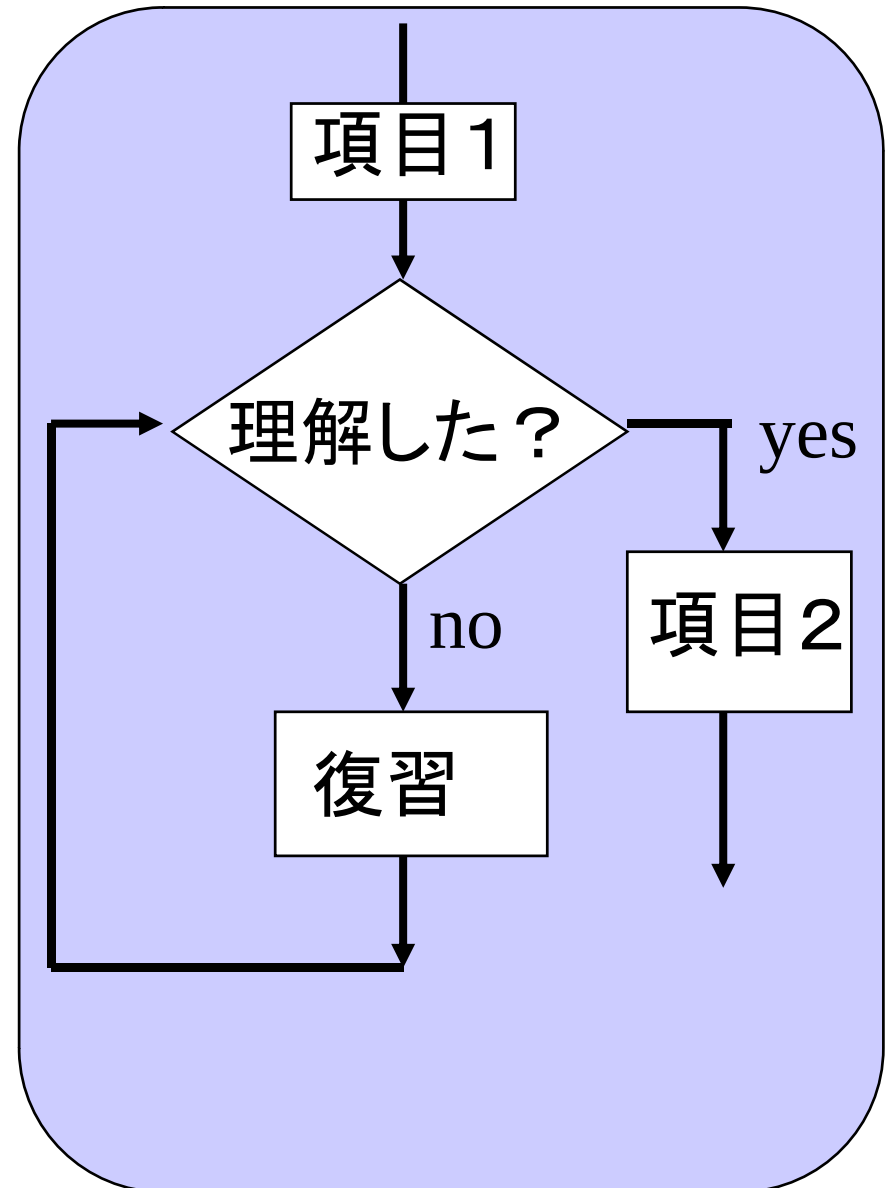
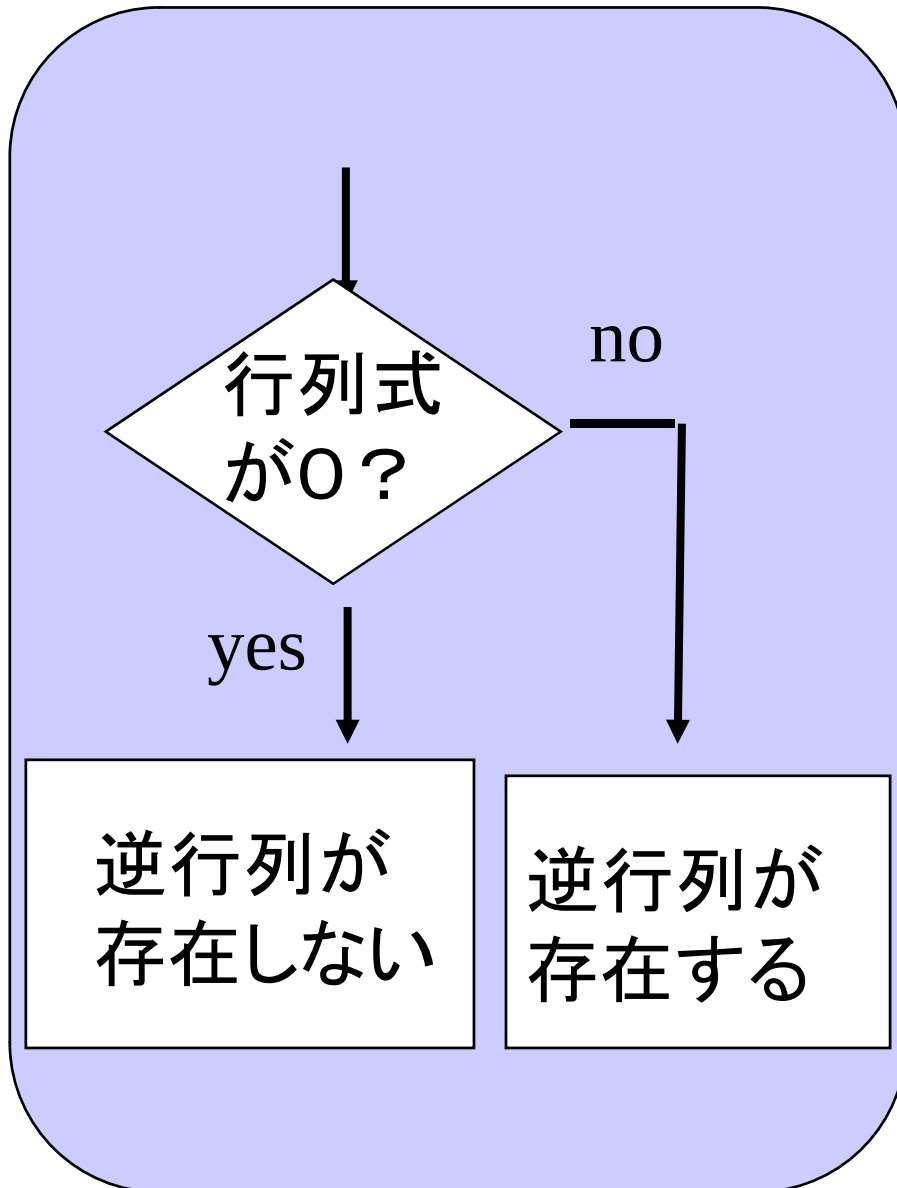
アルゴリズムの基本要素

1. ひとまとまりの処理
2. 条件判断による分岐
3. 繰り返し

フローチャート

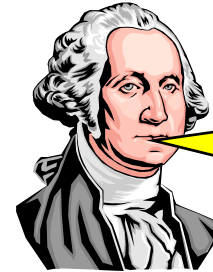
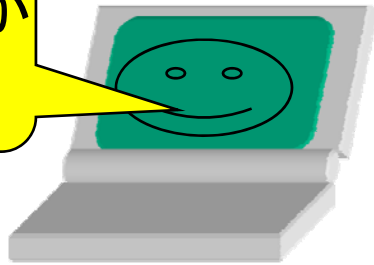


フローチャート例



プログラミング言語の分類 (高級言語と低級言語)

0と1の列しか
わからない



日本語しか
わかんない

コンピュータ側

機械語

アセンブリ言語

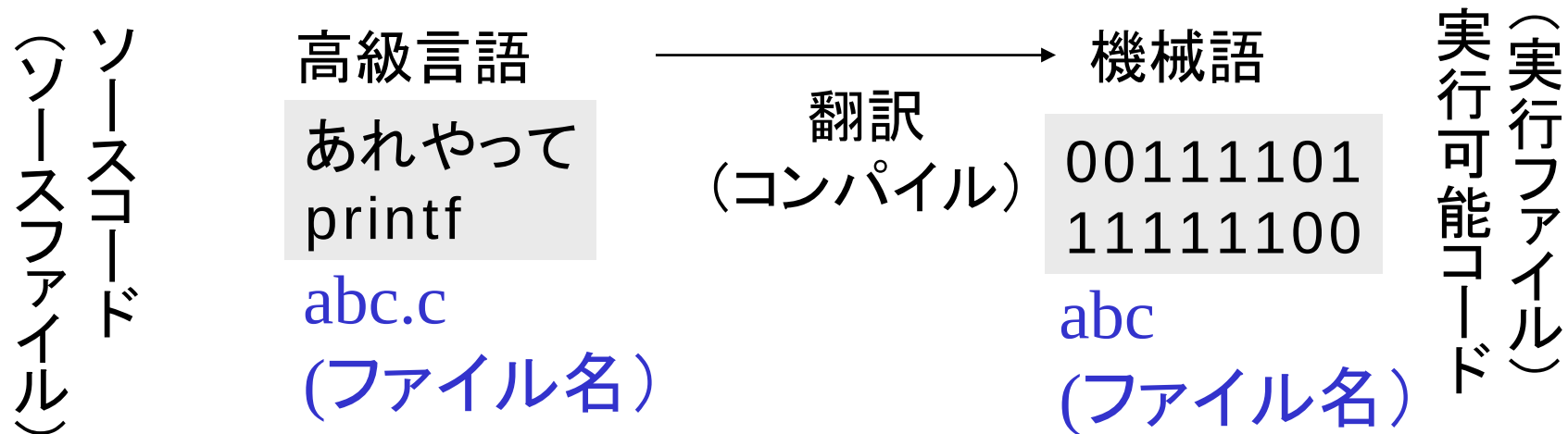
← 低級



人間側

→ 高級

コンパイラ



コンパイラ:

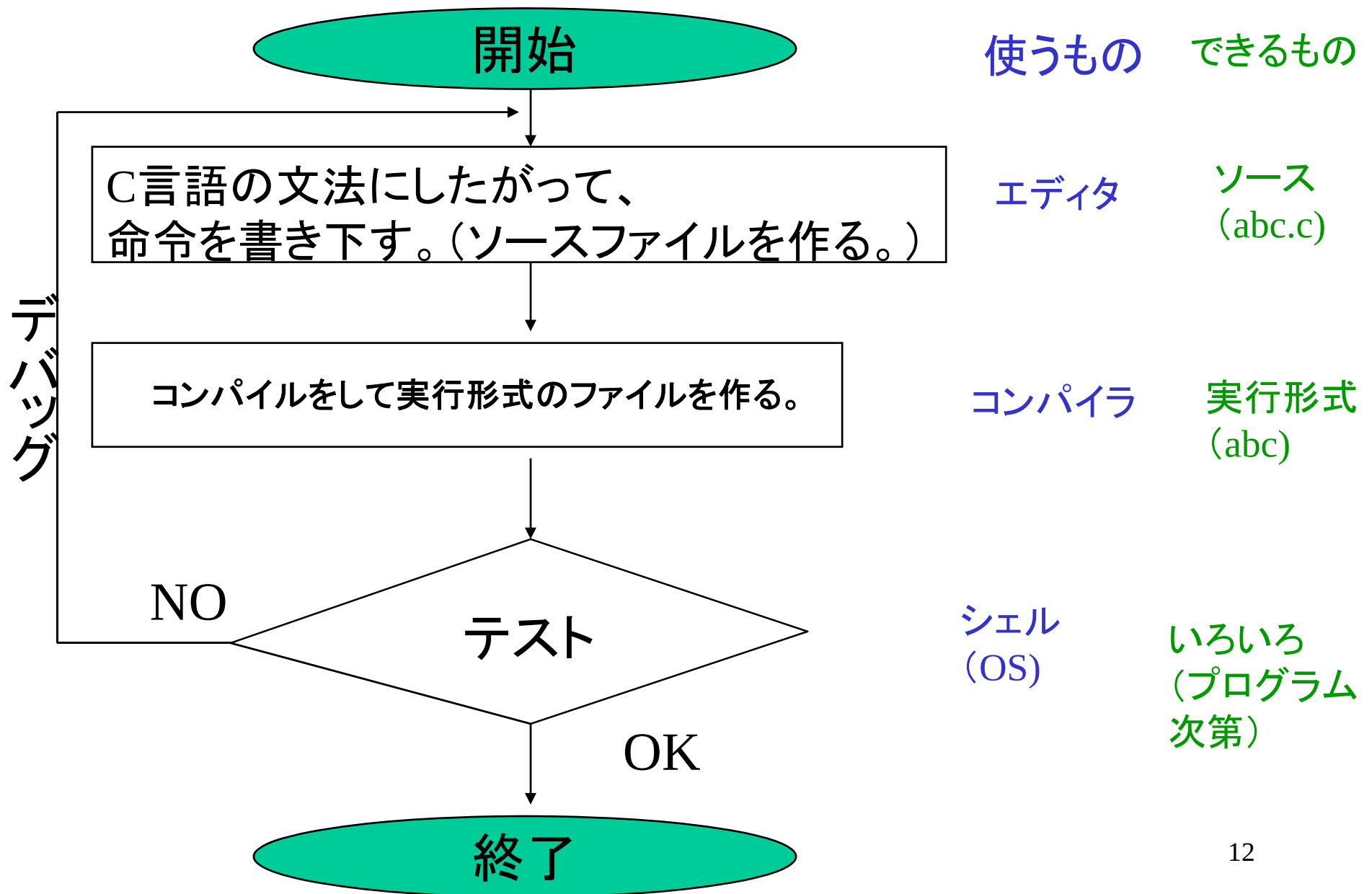
高級言語(例えばC言語)から、低級言語(機械後)へ
翻訳(コンパイル)するソフトウェア。

注意: C言語では、ソースファイルは

*****.c

という名前にする。(拡張子が".c"のファイルにする。)

C言語でのプログラムの作り方



良いソフトウェアの基準

- 速い

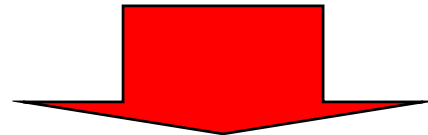
同じことするなら速い方がいいでしょ。

- 強い

どんな入力でもきちんと動作してほしいでしょ。

- 分かりやすい

誰がみても理解しやすいほうがいいでしょ。



本演習では、独自のスタイル規則に沿って
プログラミングしてもらいます。(ガイダンス資料参照)₁₃