

## 計算機言語

### 質問・相談等

24号館5階固体理論研究室  
垣谷公德

### 注意

月曜日の代講義日(土曜日)に注意

### 単位認定

レポートと最終課題、定期試験により行う

## 講義の目的

手続き型構造化プログラミング言語(FORTRAN, c, Pascal)を通じて、プログラミング言語に共通の概念、プログラミングに必要な知識を習得する

## 関連資格と講義内容

### 基本情報技術者試験

コンピュータ科学基礎  
内部データの表現  
データ構造  
コンピュータシステム  
プログラム言語  
言語処理プログラム

## 講義資料

### Webで公開中

<http://www.ee.ous.ac.jp/>

<http://sstfs.frontier.ous.ac.jp/lecture/lecture.html>



## プログラム言語の分類

### 汎用プログラム言語

### 第四世代言語

### スクリプト言語

### 特殊問題向き言語

コンピュータ上で動作するプログラムは、プログラム言語で書かれており、処理する内容によって使い分けられる。

## 汎用プログラム言語

### 低水準言語

機械語

アセンブリ言語(アセンブラ)

### 高水準言語

手続き型言語

非手続き型言語

構造化言語

オブジェクト指向型言語

## 第四世代言語

### 4th generation language (4GL)

高水準言語(第3世代言語)よりさらに抽象度の高い命令をもつプログラム言語

- 1) 高生産性、習得が容易
- 2) データベース機能を標準的にサポート
- 3) 環境からの独立性
- 4) 非手続き型

4GLの指す内容は文脈によって異なるので注意が必要

## スクリプト言語

### エンドユーザ向き言語

小規模で簡単な処理に向く

表計算のマクロ言語

Webのサービスプログラム(cgi)に利用

### 代表的なスクリプト言語

perl, ruby, python (独立汎用型)

awk, sed (目的別unix系プログラム)

sh, csh (バッチ処理型インタープリタ)

# 特殊問題向き言語

## シミュレーション言語

GPSS, DYNAMO

## 数値制御用言語

APT

## 土木建築用言語

COGO, STRESS

## 統計処理用言語

SAS

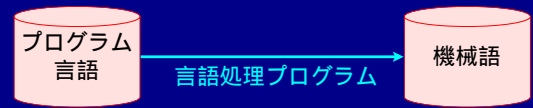
# 言語処理プログラム

## 機械語

通常唯一の命令列

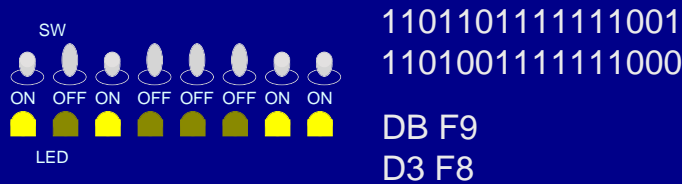
通常0/1の2進数で表現される

人間に理解し易いように16進数が使われることもある



## 機械語

実験用マイクロコンピュータボードで  
入力SWに対応するLEDを点灯する



ただし、入力SWのポート番号は11111001(F9)  
出力LEDのポート番号は11111000(F8)とする

## 低水準言語

### 機械語

### アセンブリ言語

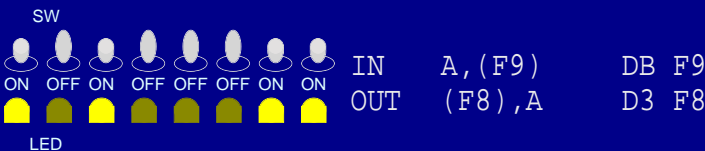
人間に解りにくい

特定のハードウェアに依存する

プログラムを小さくすることができる

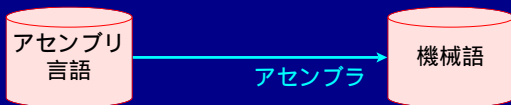
プログラムを高速に実行することができる

## アセンブリ言語



IN A, (??) DB ??  
OUT (??), A D3 ??

機械語と一対一に対応



入力SWのポート番号は(F9)、出力LEDのポート番号は(F8)、  
プロセッサはZ80

## 高水準言語

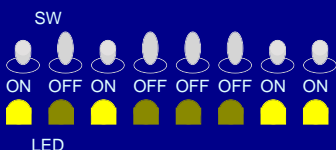
人間の言語表現に近い構造をもつ  
記述に、汎用性・可搬性がある

特定のハードウェアに依存しない

### 現在の主流

COBOL, Fortran, c, pascal, c++, java  
lisp, prolog, SQL

## 高水準言語



### c言語

```
unsigned int a;  
a = get_SW_status();  
set_LED_on(a);
```

### Fortran

```
INTEGER A  
READ (9, *) A  
WRITE (8, *) A
```

入出力関数は専用ライブラリで  
用意されなければならない

入出力装置番号8/9はシステム  
により定義されていない  
ならない

## コンパイラとインタプリタ

### コンパイラ(翻訳型)

高水準言語で書かれたプログラムを機械語  
に一括翻訳した後、実行する

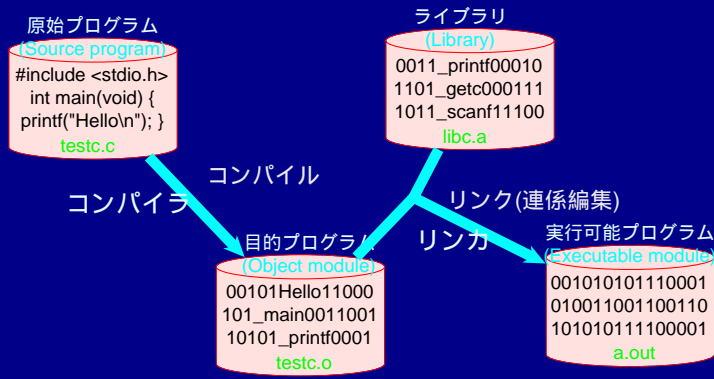
高速、エラーを取り除くのが難しい

### インタプリタ(解釈型)

高水準言語で書かれたプログラムを一行づ  
つ機械語に解釈し逐次実行する

低速、扱い易い

# プログラムの作成から実行まで

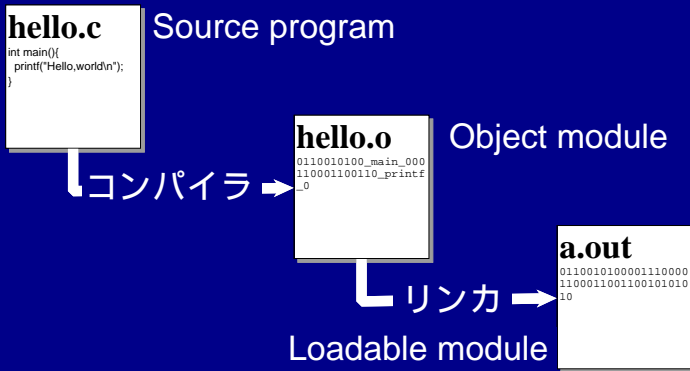


# プログラム作成手順

## 概要

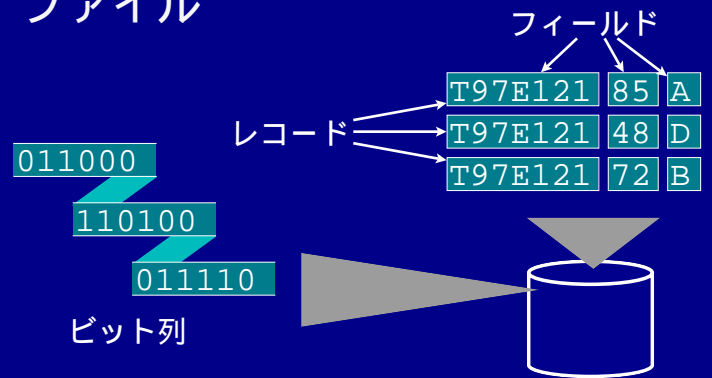
- ・ 原始プログラム(Source program)の作成
- ・ 原始プログラムをコンパイルし目的プログラム(Object module)に変換
- ・ 目的プログラムを連係編集(link)し、実行可能プログラム(Executable module/Load(able) module)に変換
- ・ 実行可能プログラムをメモリ上にロードし実行する

# プログラムの作成から実行まで



# プログラム(モジュール)の保存

## ファイル



# プログラム作成用語集(1)

## 原始プログラム

source program/source code/source file  
プログラム言語によって記述されたもの

## 目的プログラム

object module/object code/object file  
コンパイラによって変換されたもの  
アドレスなどが未定義のため実行不可

# プログラム作成用語集(2)

## 実行可能プログラム

executable file/load module/command  
プログラムの実行に必要なサブルーチンや関数などを含めた計算機で実行可能なもの

## リンク(連係編集)

目的プログラムのサブルーチン、関数のアドレス定義を行いライブラリなどと結合させて実行可能プログラムを作成する  
リンカージェディタ(リンカ)によって行う

# コンパイル

## 概要

原始プログラムを機械語に翻訳する作業

## 処理過程

- (1) 字句解析
- (2) 構文解析
- (3) 意味解析
- (4) 最適化
- (5) コード生成

# アプリケーションの種類

## GUI アプリケーション

(GUI: Graphical User Interface)  
高水準言語からOSの提供するAPIを呼び出すことにより作成する

## コンソールアプリケーション

OSの提供する標準入出力を使用して作成し、CUIから利用する  
(CUI: Character User Interface)

## 主なプログラム言語

### Fortran

最初の高水準言語、科学技術計算に向く

### COBOL

主に事務処理用に用いられる

### c言語

unixの標準開発環境、汎用性が高い

### Pascal

プログラミング教育用

## 主なプログラム言語

### C++

c言語にオブジェクト指向の機能を付加

### LISP

関数型言語、人工知能やリスト処理用

### Smalltalk

最初のオブジェクト指向言語

### Java

高可搬性オブジェクト指向インタプリタ