

# 基礎プログラミングI

---

## 第9回 計算機の内部表現

メディア情報コース  
平居 悠（ひらい ゆたか）

# 到達目標

---

## 計算機上での情報の取り扱い方の基礎の理解

1. コンピュータに指示を送る基本を理解する。
2. 文字列や数値を基本とした値の概念とそれを格納する変数の仕組みを理解する。
3. 変数一つで集合を表す概念を理解する。
4. 2進数を基本としたコンピュータの内部表現について基本を理解する。
5. 現実社会の簡単な問題を扱うプログラムを作成できるようになる。

# 前回

---

第1回	4月9日	プログラミングの基礎
第2回	4月16日	変数・制御構造
第3回	4月23日	メソッド、値の型変換
第4回	4月30日	確認テスト
第5回	5月14日	集合処理1（配列）
第6回	5月21日	集合処理2（CSVとデータ処理）
第7回	5月28日	集合処理3（ハッシュ）
<b>第8回</b>	<b>6月11日</b>	<b>正規表現</b>
第9回	6月18日	計算機の内部表現
第10回	6月25日	スタイルとデバッグ、実用的なプログラム
第11回	7月2日	作品の公開
第12回	7月9日	チーム課題作成
第13回	7月16日	チーム課題発表
第14回	日程未定	期末試験

# 前回の目標

---

正規表現を利用したデータの  
フィルタリングができる

# まとめ

---

- データから何かを検索するときに、**正規表現**を用いる。
- Rubyでは、正規表現を**/パターン/**と表記する。
- **メタキャラクタ**を用いて検索方法を指定することができる。

# アイスブレイク

---

同じチームの人に1人1分程度で自己紹介をしよう。その際、「#07 事前課題 データCSV作成」でどのようなデータを作成したか説明しよう。

# タイピング練習スケジュール

---

- 第1回 ホームポジション
- 第2回 ローマ字
- 第3回 英語初級
- 第4回 日本国憲法 (trr試験)
- 第5回 ホームポジション
- 第6回 ローマ字
- 第7回 英語初級
- 第8回 日本国憲法 (trr試験)
- 第9回 ホームポジション
- 第10回 ローマ字
- 第11回 英語初級
- 第12回 日本国憲法 (trr試験、合格スコア150)

# trr起動方法

---

1. ブラウザを起動し、<https://www.koeki-prj.org/trr/>に繋ぐ。
2. 学籍番号（Cは大文字、省略なし8桁）を入力する。
3. Koeki MAILに届いたパスコードをPasscode: 欄に入力する。

# ホームポジション

左手でタイプするキー

右手でタイプするキー



左手の人差指から小指までのホームポジション

両手の親指のホームポジション

右手の人差指から小指までのホームポジション

# 今回

---

第1回	4月9日	プログラミングの基礎
第2回	4月16日	変数・制御構造
第3回	4月23日	メソッド、値の型変換
第4回	4月30日	確認テスト
第5回	5月14日	集合処理1（配列）
第6回	5月21日	集合処理2（CSVとデータ処理）
第7回	5月28日	集合処理3（ハッシュ）
第8回	6月11日	正規表現
<b>第9回</b>	<b>6月18日</b>	<b>計算機の内部表現</b>
第10回	6月25日	スタイルとデバッグ、実用的なプログラム
第11回	7月2日	作品の公開
第12回	7月9日	チーム課題作成
第13回	7月16日	チーム課題発表
第14回	日程未定	期末試験

# 今回の目標

---

文字コードと画面表示系の  
仕組みを理解する

# 事前課題試問

---

事前課題で提出した以下のテーマのプログラムと説明資料に関してTAに説明せよ。

1. gets, to\_i, to\_f, to\_s と printf の %d, %s, %x
2. if/while/break と 真偽値
3. 配列と添字, <<, length, each と for による繰り返し
4. CSV ファイルの構造と作成方法・作成時の注意
5. ヘッダ付き CSV ファイルの Ruby プログラムからの処理の仕方
6. 正規表現とはどういうものでどういうときに使うものか
7. 正規表現の特殊文字の使い方 . [] ? + \* () \$1, \$2, \$3, ...

# Rubyプログラムでの文字コードの扱い

ある文字コードを画面に出力したい場合は、`printf`のフォーマット文字列に%cを指定する。

```
printf(“%c”, 0x07)
```

とすると、文字コード0x07つまりBELが出力され、端末のベルが鳴る。irbを起動し、上記を実行してみよう。

# Rubyプログラムでの文字コードの扱い

ある文字の文字コードを得るには**?文字.ord**を使う。

ASCIIコード表の  
“!” (0x21)–“z”  
(0x7A)の範囲の  
文字全てを出力  
するプログラム  
ascii.rbを書いて  
実行してみよう。

```
#!/usr/koeki/bin/ruby
# -*- coding: utf-8 -*-

char = ?!.ord  # !の文字コード(==0x21)

while char <= ?Z.ord
  printf("ascii-code %3d(0x%02x): %c\n",
    char, char, char)
  char += 1
end
```

# 文字列と文字コード

文字列は、あたかも文字コードの数値が並んで入っている配列のようなものである。

<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
H	e	l	l	o
0x48	0x65	0x6c	0x6c	0x6f

# 文字列と文字コード

文字列“Hello”の最初の部屋 (0号室)を取り出すときは、配列のときと同様[ ]を利用する。

```
printf("%c\n", "Hello"[0])
```

↓

H

```
printf("%x\n", "Hello"[0].ord) (.ordにより文字コードが得られる)
```

↓

48 (これは16進数。以下も同じ)

# 文字列と文字コード

文字列の先頭から末尾まで繰り返すプログラム `hellocode.rb` を書いて実行してみよう。

```
#!/usr/koeki/bin/ruby
# -*- coding: utf-8 -*-

hello = "Hello"
i = 0

while i < hello.length
  printf("%c → 0x%x\n", hello[i], hello[i].ord)
  i += 1
end
```

# 課題

---

以下の各問について解答し、s4基礎プロIG、Hの「#09 提出課題」に書き込む。

1. 自分の誕生日の西暦年、月、日を全部足した数を求め、2進数と16進数に直せ。
2. ローマ字で書いた自分の名前（下の名前）の文字列の各文字の文字コードを16進数で表記せよ。16進数はつなげずに文字ごとに区切って記すこと。

締め切り：6月22日(月)

# 今回の目標

---

文字コードと画面表示系の  
仕組みを理解する

# まとめ

---

- 計算機はデータを**2進数**の状態です保存する。
- 計算機内では文字に付された番号（**文字コード**）を用いて文字を扱う。
- 文字列を表現する方式のことを**エンコーディング**という。

# 次回

---

第1回	4月9日	プログラミングの基礎
第2回	4月16日	変数・制御構造
第3回	4月23日	メソッド、値の型変換
第4回	4月30日	確認テスト
第5回	5月14日	集合処理1（配列）
第6回	5月21日	集合処理2（CSVとデータ処理）
第7回	5月28日	集合処理3（ハッシュ）
第8回	6月11日	正規表現
第9回	6月18日	計算機の内部表現
第10回	6月25日	スタイルとデバッグ、実用的なプログラム
第11回	7月2日	作品の公開
第12回	7月9日	チーム課題作成
第13回	7月16日	チーム課題発表
第14回	日程未定	期末試験

# 事前課題

---

1. 第1回授業資料「間違いの修正」

<https://www.koeki-prj.org/~yuuji/2026/pf1/02/correction.html>  
を読む。

2. 第10回授業資料「スタイルとデバッグ」

<https://www.koeki-prj.org/~yuuji/2026/pf1/debugging/index.html>  
「周辺知識」

<https://www.koeki-prj.org/~yuuji/2026/pf1/10/misc.html>を読む。

# 事前課題

---

3. 第10回講義ノート「楽しいプログラムのための部品」

<https://www.koeki-prj.org/~yuuji/2026/pf1/10/tricks.html>

を参考にし、

- 5桁以上の出力を所々色分けして出力するプログラム `colormsg.rb`

または

- 画面に自分で作った画像を表示するプログラム `showning.rb`  
いずれかを作成せよ。

# レポート提出方法

---

s4基礎プロIG, Hの「#10 事前課題 color/img」に以下の通り書き込む。

1. クラス、学籍番号、氏名、共同作成者 (チームメンバー)
2. (色を変えるor画像表示する)方法の実例を含めた説明
3. 作成プログラムと結果画面のスクリーンショット画面

# レポート提出方法

---

画像はターミナルで

```
import myscreen.jpg
```

のようにして、そのターミナル画面または背景画面（画面全体を撮りたい場合）をクリックすることで得られる。画像を確認したい場合は`display myscreen.jpg`とする。

**締め切り：6月24日(水)**