

3 プログラムの構成要素①

■ プログラムとは何か、プログラムの構成要素には何があるかを学ぼう。

調べてみよう

プログラムという言葉の意味を辞書で調べてみよう。

1 プログラムとは

アルゴリズムをコンピュータが処理できるように、プログラミング言語で記述したものをプログラムという。

ハードウェアであるコンピュータは、プログラムがなければ動作しない。いわば、プログラムはコンピュータに対する指示書である。

例 鉄道路線の駅番号から駅名を検索する

```

1 stations = ['天王寺', '寺田町', '桃谷', '鶴橋',
              '玉造', '森ノ宮', '大阪城公園', '京橋',
              '桜ノ宮', '天満', '大阪', '福島',
              '野田', '西九条', '弁天町', '大正',
              '芦原橋', '今宮', '新今宮']
2
3 print('駅番号を入力してください')
4 station_number = input()
5 number = int(station_number)
6
7 if number > 0 and number < 20:
8     print(stations[number - 1], '駅です')
9 else:
10    print('該当する駅はありません')
```

プログラムはどれも Python (パイソン) というプログラミング言語で説明しているよ

変数 p.135

変数名のつけ方には明確な決まりはないが、特別な文字列は使用できない(p.160参照)。

データ型 p.135

記述した値が文字なのか数値なのかをコンピュータが判別できるようにデータ型を定義してプログラムを作成する必要がある。

関数(組み込み関数) p.136

print関数は文字列を出力する。
input関数はキーボードから入力された文字列を受け取る。
int関数は文字列としての数字を整数値に変換する。

演算 p.135

プログラム内で演算をあらわす記号を演算子という。

データ構造 p.138

stations[]は「配列」を示している。配列の要素は0番目からはじまるため、実際に入力した値とは1ずれることに注意する。Pythonにおける配列の扱いについては、p.161を参照。

実行結果

駅番号を入力してください
11
大阪 駅です

赤字はユーザが入力した数値。入力値「11」に対応する「大阪」が表示される。

ここでは基本的なプログラムがどんな要素で構成されているかさえ理解できればOKよ

if~elseが前に習った「分岐構造」(p.132参照)を表現しているんだ

2 プログラムの構成要素

プログラムは、基本的に次のような要素によって構成されている。

● 変数

変数とは、名前のついた箱

のようなもので、文字列や数値などの値を格納するために

使う。変数に値を設定し、そ

の変数から値を読み込んだり書き換えたりすることで、データを効率的に扱える。変数に値を設定することを代入という。

演算例

a = 2	←変数aに2を代入
b = 3	←変数bに3を代入
a + b	←2 + 3を実行
5	←実行結果

● データ型

扱うデータがどのような性質のもので、どのように取り扱うべきかを定めたものをデータ型という。たとえば、

1を数値として扱うか、文字列として'1'のように扱うかで演算した場合の結果は異なる。数値であれば「1 + 1」の演算結果は2になるが、文字であれば「'1' + '1」の演算結果は、1を2つ並べた「11」になる。プログラミング言語によってデータ型の扱いは異なるが、意図した結果を得るためにはデータ型の定義が重要になる。

演算例

1 + 1	←1 + 1を実行
2	←実行結果
'1' + '1'	←'1'と'1'を結合
11	←実行結果

● 演算

演算には、四則演算など数値を計算する算術演算、2つの値の大小関係や等値関係を判定する比較演算、複数の真値を組み合わせたたりするなどの論理演算、変数に値を代入するための代入演算などがある。

プログラム内で演算をあらわす記号を演算子といい、た

たとえば、1と2の和をあらわす「1+2」では「+」がその演算子に当たる。

演算例

a = 5	←変数aに5を代入
b = 3	←変数bに3を代入
a / b	←5 ÷ 3を実行
1.6666666666666667	←実行結果
a ** b	←5 ³ を求める
125	←実行結果
a == b	←5と3は等しい
False	←実行結果
(a > b) or (a < b)	←5 > 3または5 < 3のいずれかは正しい
True	←実行結果

ここであげられた要素は多くのプログラミング言語で共通しているよ



1 変数

variable

変数を使う際、あらかじめ変数名やデータ型を宣言する必要のあるプログラミング言語もある。

◆ データ型の種類

プログラミング言語によって定義は異なるが、おもなデータ型には次のようなものがある。

数値	
整数型	整数をあらわす型
固定小数点型	小数点の位置をあらかじめ決めてあらわす型
浮動小数点型	小数点の位置を、浮動小数点の指数部で指定する型

その他

文字列型	文字列をあらわす型
論理値型	真値の真(True)と偽(False)をあらわす型

やってみよう

前ページのプログラムを身近な交通機関の駅名に置き換えてみよう。

3 プログラムの構成要素①

■ プログラムとは何か、プログラムの構成要素には何があるかを学ぼう。

調べてみよう

プログラムという言葉の意味を辞書で調べてみよう。

1 プログラムとは

アルゴリズムをコンピュータが処理できるように、プログラミング言語で記述したものをプログラムという。

ハードウェアであるコンピュータは、プログラムがなければ動作しない。いわば、プログラムはコンピュータに対する指示書である。

例 鉄道路線の駅番号から駅名を検索する

```

1  stations = ['天王寺', '寺田町', '桃谷', '鶴橋',
              '玉造', '森ノ宮', '大阪城公園', '京橋',
              '桜ノ宮', '天満', '大阪', '福島',
              '野田', '西九条', '弁天町', '大正',
              '芦原橋', '今宮', '新今宮']
2
3  print('駅番号を入力してください')
4  station_number = input()
5  number = int(station_number)
6
7  if number > 0 and number < 20:
8      print(stations[number - 1], '駅です')
9  else:
10     print('該当する駅はありません')
```

実行結果

```

駅番号を入力してください
11
大阪 駅です
```

赤文字はユーザが入力した数値。入力値「11」に対応する「大阪」が表示される。

ここでは基本的なプログラムがどんな要素で構成されているかさえ理解できればOKよ

if~elseが前に習った「分岐構造」(p.132参照)を表現しているんだ

プログラムはどれも Python (パイソン) というプログラミング言語で説明しているよ

変数 p.135

変数名のつけ方には明確な決まりはないが、特別な文字列は使用できない(p.160参照)。

データ型 p.135

記述した値が文字なのか数値なのかをコンピュータが判別できるようにデータ型を定義してプログラムを作成する必要がある。

関数(組み込み関数) p.136

print 関数は文字列を出力する。
input 関数はキーボードから入力された文字列を受け取る。
int 関数は文字列としての数字を整数値に変換する。

演算 p.135

プログラム内で演算をあらわす記号を演算子という。

データ構造 p.138

stations[] は「配列」を示している。配列の要素は 0 番目からはじまるため、実際に入力した値とは 1 ずれることに注意する。Python における配列の扱いについては、p.161 を参照。

① デジタルデータと2進法

- ☑ アナログとデジタルの特徴を理解しよう
- ☑ 2進法による表現を理解しよう



■ アナログとデジタル

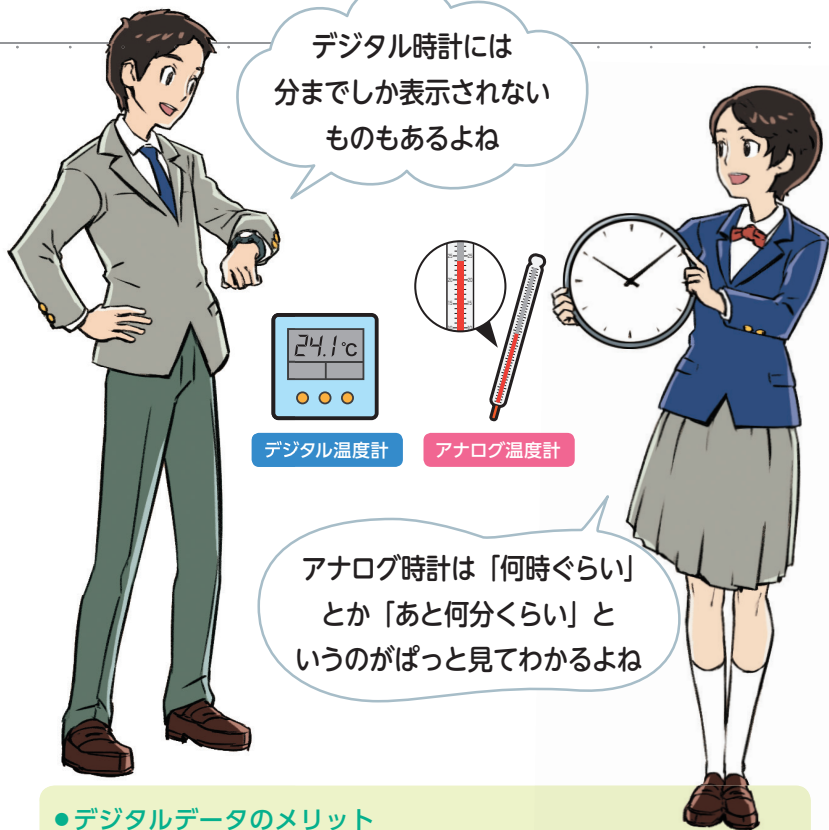
コンピュータで情報を表現できるのは、情報がデジタル化されているからである。デジタル化とは、文字、音、画像、動画などの情報を数値であらわすことである。

アナログ

連続的に変化する量を別の連続する量で表現する方式。たとえば身長は柱に目印をつけて「このくらい」と表現できる。アナログ時計は針の角度で時刻をあらわしている。

デジタル

変化する量を一定の間隔で区切って数値で表現する方式。身長は身長計の目盛りを読み取って173.4cmのような数値で表現できる。時刻も9時30分12秒のような数値で表現できる。

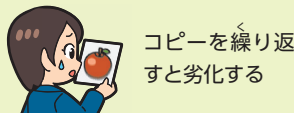


● デジタルデータのメリット

アナログ



デジタル



スマホ1台で、写真や音楽、友だちとやり取りしたたくさんのメッセージも扱えるのは、情報がデジタル化されているからなんじゃよ!



■ 2進法による表現と情報量の単位

10進法の数の意味

1	2	3	4
---	---	---	---

10^3 (千) の位	10^2 (百) の位	10^1 (十) の位	10^0 (一) の位
---------------------	---------------------	---------------------	---------------------

$$1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

$$= 1000 + 200 + 30 + 4$$

$$= 1234$$

2進法→10進法

1	1	0	1
---	---	---	---

2^3 の位	2^2 の位	2^1 の位	2^0 の位
-------------	-------------	-------------	-------------

$$1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

$$= 8 + 4 + 0 + 1$$

$$= 13$$

この1桁が1ビットね



2進法の1101を10進法であらわすと

0乗は1になるよ

10進法・2進法・16進法の関係

10進法	2進法	16進法
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10
⋮	⋮	⋮

10進法→2進法

10進法の11を、2進法であらわすと

$$11_{(10)}$$

$$\begin{array}{r} 2 \overline{) 11} \\ \underline{2} \\ 2 \\ \underline{2} \\ 0 \\ \underline{0} \\ 0 \\ \underline{0} \\ 0 \end{array}$$

2で割る

余り

1011₍₂₎ となる

2進法→16進法

$$00101011_{(2)}$$

4ビットずつに区切る

$$\begin{array}{cc} 0010 & 1011 \\ = 2_{(16)} & = B_{(16)} \end{array}$$

$$= 2B_{(16)}$$

16進法→2進法

$$6A_{(16)}$$

それぞれの桁を2進法にする

$$\begin{array}{cc} 6_{(16)} & A_{(16)} \\ = 110_{(2)} & = 1010_{(2)} \end{array}$$

$$= 1101010_{(2)}$$

● 情報量の単位と大きさ

単位	読み方	大きさ
bit	ビット	—
B	バイト	1B=8bit
KB	キロバイト	1KB=1024B
MB	メガバイト	1MB=1024KB
GB	ギガバイト	1GB=1024MB
TB	テラバイト	1TB=1024GB
PB	ペタバイト	1PB=1024TB

2¹⁰=1024倍ごとに変わるんだね



やってみよう!

- 身近にあるアナログ表現のものをあげてみよう
- 身近にあるデジタル表現のものをあげてみよう

コンピュータは計算する機械である。すべての情報はデジタル化され、0と1だけのデータとして扱われる。この0と1だけで数をあらわす方法を2進法という。情報量をあらわす最小単位のビットは、2進法の1桁に当たる。8ビットをひとまとまりにしたバイトという単位も情報量の単位である。

進法変換

2進法は桁の数が多くなり人が読むには紛らわしい。そのためプログラミングなどでは16進法も使われる。わたしたちが普段使う10進法とは異なるが、あらわしている量や数は変わらないため、相互に変換できる。その数が何進法かを示したいときは左図のようにカッコつきの添え字を使い、1011₍₂₎、11₍₁₀₎、B₍₁₆₎のように記す。

ビット

情報量の最小単位。1ビットは0か1かの2通りの状態をあらわすことができる。

バイト

8ビットをひとまとまりにした情報量の単位。1バイトでは2の8乗、0~255までの256通りの状態をあらわすことができる。

接頭語

ファイルサイズなどはバイト (B) であらわされる。大きなサイズの場合は、キロ (K) やメガ (M) などの接頭語を用いてあらわす。

① デジタルデータと2進法

- アナログとデジタルの特徴を理解しよう
- 2進法による表現を理解しよう



■ アナログとデジタル

コンピュータで情報を表現できるのは、情報が**デジタル化**されているからである。デジタル化とは、文字、音、画像、動画などの情報を数値であらわすことである。

アナログ

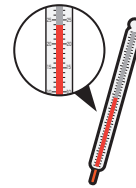
連続的に変化する量を別の連続する量で表現する方式。たとえば身長は柱に目印をつけて「このくらい」と表現できる。アナログ時計は針の角度で時刻をあらわしている。

デジタル

変化する量を一定の間隔で区切って数値で表現する方式。身長は身長計の目盛りを読み取って173.4cmのような数値で表現できる。時刻も9時30分12秒のような数値で表現できる。



デジタル温度計



アナログ温度計

アナログ時計は「何時ぐらい」とか「あと何分ぐらい」というのがぱっと見てわかるよね

デジタル時計には分までしか表示されないものもあるよね



● デジタルデータのメリット

アナログ



書いたり消したりがたいへん

デジタル



データの加工や編集が簡単



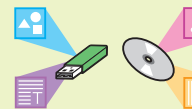
コピーを繰り返すと劣化する



複製による劣化がない



それぞれに適したメディアが必要

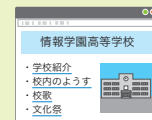


同じ記録メディアに保存できる

スマホ1台で、写真や音楽、友だちとやり取りしたたくさんのメッセージも扱えるのは、情報がデジタル化されているからなんじゃよ!



文字・画像・音声・動画などが別々



異なる種類のデータを統合できる

ペーパープロトタイピングを してみよう

知ってる？ スマホのアプリをつくる時って、画面や動きを確認するプロトタイプをつくるんだって！



きっと、アルゴリズムやプログラムを考えていくのにいい方法なのね！

テーマ & 目標

- アプリを設計し、ペーパープロトタイピングでその動きを表現する

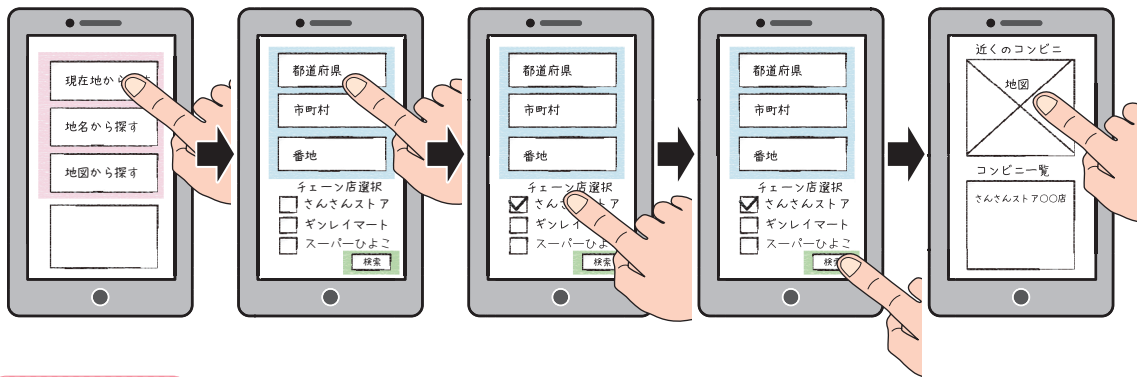
実習 アプリを企画し、ペーパープロトタイピングをしてみよう

自分が使ってみたいアプリを企画しよう。アイデアが固まったらプロトタイプをつくり、その画面や動きを確認してみよう。最後に、完成したプロトタイプを使って発表しよう。

実習の流れ



手順1 まず、ペーパープロトタイピングとはどのようなものかを知ろう。



ONE POINT

ペーパープロトタイピングとは

- 紙でアプリやWebサイトの画面や動きのイメージを表現すること。
- 画面を紙や付せんなどで複数枚つくって、紙芝居のようにアプリの動きを表現し確認する。
- 実際にアプリ開発の現場でも使われている手法。

手順2 どのようなオリジナルアプリをつくるか企画しよう。

【1】 「こんなアプリが欲しい」というアイデアを、まずは自分で考えてメモしよう。

【2】 3～4人のグループで、互いの企画案について意見交換をしアイデアを共有しよう。友だちからもらったアドバイスや、話し合いの中で出てきたアイデアは、下にメモしておこう。

【3】 【2】のメモを参考に、自分のアイデアを固めて、企画書をつくろう。

企画書

タイトル	例 食事のカロリーを教えてくれるダイエットアプリ。
ターゲット	例 カロリー摂取量が気になる若い男女向け。
アプリの概要	例 スマートフォンで食べ物の写真を撮影すると、自動的にカロリーを表示してくれる。
提案の理由	例 食事のカロリーが気になる人は多いと思ったから。

ペーパープロトタイピングを してみよう

知ってる？ スマホのアプリをつくるときって、画面や動きを確認するプロトタイプをつくるんだって！



きっと、アルゴリズムやプログラムを考えていくのにいい方法なのね！

テーマ & 目標

- アプリを設計し、ペーパープロトタイピングでその動きを表現する

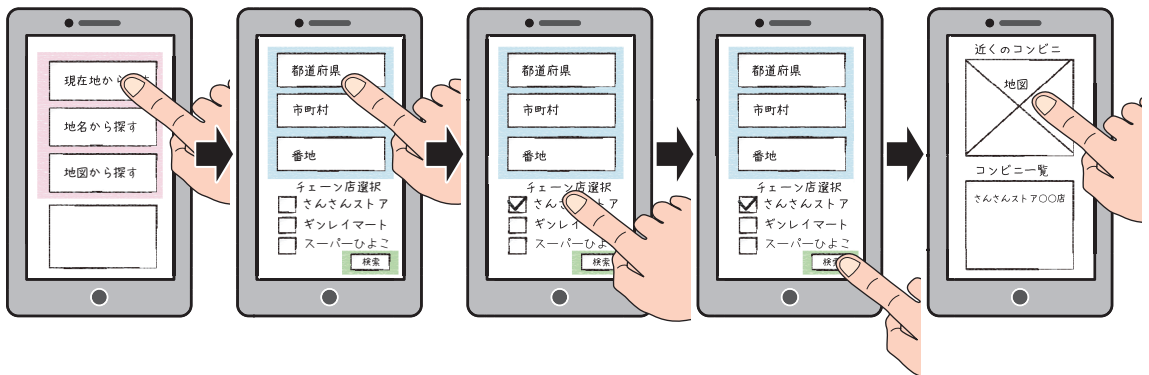
実習 きかく アプリを企画し、ペーパープロトタイピングをしてみよう

自分が使ってみたいアプリを企画しよう。アイデアが固まったらプロトタイプをつくり、その画面や動きを確認してみよう。最後に、完成したプロトタイプを使って発表しよう。

実習の流れ



手順 1 まず、ペーパープロトタイピングとはどのようなものかを知ろう。



ONE POINT

ペーパープロトタイピングとは

- 紙でアプリや Web サイトの画面や動きのイメージを表現すること。
- 画面を紙や付せんで複数枚つくって、紙芝居のようにアプリの動きを表現し確認する。
- 実際にアプリ開発の現場でも使われている手法。