

## 第4回 計算プログラム

入力された値に対して計算をするプログラムを作成する。

### 【入力を受け付けるプログラム】

★足し算をするプログラムを作成して実行しましょう。Python3 を選ぶのを忘れずに。

```
a, b = map(int, input().split())
c = a + b
print(c)
```

今回は「入力を受け付けるプログラム」なので、前回と同様にプログラムを打ち込んですぐに「実行」ボタンを押すと、次のようなエラーが表示される（ターミナルからプログラムのファイルを実行すると、入力待ちの状態となる）。

```
Traceback (most recent call last):
  File "Main.py", line 1, in <module>
    a = int(input())
ValueError: invalid literal for int() with base 10: ''
```

Paiza では、ユーザーの入力を「入力」欄に入れることができる。ここでは、「入力」欄に次のように 2 つの数を入れる。

```
23 35
```

そして「実行」ボタンを押すと、出力には入力した 2 つの数 23 と 35 を足した「58」が計算結果として表示される。このように、入力した 2 つの数を合計した数を出力するプログラムを書いたことになる。

エラーが出る場合には、前回の授業のエラーの修正方法を確認して修正すること。正常に実行されたら、入力欄に他の数（整数）を入れて試してみること。

### 【入力の読み取り】

このプログラムでは、`input` 関数によって入力を読み取っている。`input()` とすることで、入力から 1 行文字列が読み取られる。

```
a, b = map(int, input().split())
```

の行は、現段階ですべてを理解するのは大変であるが、簡単に解説をしておく。とりあえずは雰囲気を感じ取れれば十分である。たとえば、「23 35」と入力された場合には、`input()` によって”23 35”という文字列が読み取られる。その後、`split()` によって、文字列がスペースで分割されたリストとなり、`map` 関数によってリストの要素にそれぞれ `int` 関数が適用されて文字列から整数に変換され、得られた整数が順番に `a` と `b` という変数に格納される。

まとめると、この行が実行されることで、「23 35」と入力したときに `a` と `b` という変数に 23 と 35 という整数が格納されることとなる。そのような「動き」がわかれば、現段階では十分である。変数の意味については、次に解説する。このような「入力を変数に格納する」コードは、今後も同様のコードを使う。

**【変数と型】**

変数は、値を入れておく箱のようなものである。数や文字などを格納することができる。Python では、変数に値が格納されるときに「データ型」<sup>1</sup>が決まる。ここでは 3 種類のデータ型について解説する。

- (1) str (文字列型) 前回の授業では “Hello!” という文字列を扱った。文字が並んでいるデータ型である。文字には「改行」などの特殊文字も含まれる。
- (2) int (整数型) 数学で定義される「整数」と同じで、1, 2, 3 のような正の整数、-1, -2, のような負の整数、そして 0 である。2.5 のような数は整数ではない。int 型にはメモリが 4 バイトのように決められて大きさの制限があるプログラミング言語が多いが、Python 3 の int 型はメモリの許す限りいくらでも大きな整数を扱うことができる。
- (3) float (浮動小数点数型) 実数を一定の精度で表現する。たとえば、2.45 のように表現される。

```
a, b = map(int, input().split())
```

の行では、int 関数によって読み取られた文字列が整数型に変換されて、a と b という変数に格納されるため、変数 a と b はいずれも整数型の変数となる。プログラミング言語によっては、変数の型を明示的に宣言する必要があるが、Python ではこのように「整数の値が代入されたから、この変数は整数型になる」といったように自動的に変数の型が決まる。

**【プログラムの解説】**

- (1) 2 つの整数を入力

「入力の読み取り」と「変数と型」で説明した通り、入力から 2 つの整数を読み取って、整数型の変数 a と b に格納する。同様の処理はこの授業で今後頻出する。

- (2) 計算の実行

c = a + b で、a + b の計算結果が c に代入される。整数同士の足し算で結果は整数となるので、c は整数型の変数となる。数学で「=」には等式（左辺と右辺が等しい）と代入（右辺の計算結果を左辺に代入する）の意味があるが、Python では「=」は必ず代入の意味となる。したがって、右辺の a+b を計算した結果が c という変数に代入される。左辺と右辺を逆に書いてはいけない。

- (3) 計算結果の表示

print 関数によって計算結果を表示する。

**【基本演算】**

このように、Python では足し算は「+」で計算できる。同様に、四則演算（加減乗除）とべき乗は、次のように計算できる。演算子はすべて ASCII 文字であることに注意。また、累乗は「^」ではないことにも注意。

1. 足し算 : +
2. 引き算 : -
3. かけ算 : \*
4. 割り算 : / (結果は浮動小数点数型 float)
5. 割り算の整数部: // (結果は整数型)
6. べき乗: \*\*
7. 計算の順序は左から、「かけ算と割り算」は「足し算と引き算」よりも先に、「累乗」はそのいずれよりも先に計算される<sup>2</sup>。

---

<sup>1</sup> <https://docs.python.org/ja/3/reference/datamodel.html#the-standard-type-hierarchy>

<sup>2</sup> <https://docs.python.org/ja/3/reference/expressions.html#operator-precedence>

8. ( ) の中は先に計算される。( ) は何重に重ねても良い。

★足し算をするプログラムを、「引き算をするプログラム」「掛け算をするプログラム」「累乗を計算するプログラム」などに変えて、いろいろな数を入れて実行しましょう。「累乗を計算プログラム」にして大きな数を入れることで、int 型には大きさの制限がないということを確認できます。

#### 【BMI 計算プログラム】

ここまでで、「入力した数に対して計算をした結果を出力する」というプログラムを作れるようになり、これだけでかなりいろいろなプログラムを作ることができる。そこで、BMI 指数を計算するプログラムを作成して実行してみよう。

★次の BMI 計算プログラムを作成して、入力に身長(cm)と体重(kg)を「172 60」のように入れて実行しましょう。

```
a, b = map(float, input().split())
c = 10000 * b / (a ** 2)
print(c)
```

これは、BMI 指数（ボディマス指数）を計算するプログラムである。BMI 指数は、体重と身長の関係から算出した、人の肥満度を表す指数である。

身長を a[cm]、体重を b[kg]としたとき、BMI は

$$\text{BMI} = 10000 \frac{b}{a^2}$$

と定義されている。日本肥満学会によると、BMI が 22 の場合が標準体重で、25 以上が肥満、18.5 未満が低体重である。BMI の定義は世界共通だが、肥満の定義は各国によって違い、男女で定義が異なる場合もある。

このプログラムを説明する。

- (1) 入力を a, b という変数に格納するところは足し算プログラムと同じである。
- (2) BMI 指数を計算して c に格納する。割り算 / が含まれるため float 型になる。
- (3) BMI 指数を表示する。

★自分の身長、体重を入れて、BMI 指数を計算してみましょう。また、身長を変えずに、体重を増減させてみて、BMI 指数がどのように変わるか計算してみましょう。次回はこのプログラムを改良する予定です。

#### 【★課題】

児童・生徒の肥満の程度をあらわすローレル指数は、身長を a[cm]、体重を b[kg]としたとき、以下の式で定義される。

$$\text{ローレル指数} = 10000000 \frac{b}{a^3}$$

べき乗の指数が 2 ではなくて 3 になっていることに注意。ローレル指数は 100 未満がやせ、130 が標準、160 以上が肥満とされる。なお、成人に対しては通常 BMI 指数が使われる。

BMI 計算プログラムの 2 行目を変えることで、ローレル指数を計算するプログラムを作成して、実行して正常に動作することを確認し、ToyoNet-ACE のレポートから提出してください。提出方法と採点基準は、前回と同様です。

ヒント：身長 140 cm、体重 36 kg のローレル指数は 131 程度です。