

仮入部員向け競プロ講習会 Beginner 編

KARASU4280, initial_d, maccha

2020 年 6 月 20 日

目 次

第 I 部	はじめに	2
1	競プロとは	2
1.1	魅力	2
1.2	AtCoder における実力評価	3
2	AtCoder の始め方	4
3	進め方	4
第 II 部	Python	5
1	はじめに	5
2	if	5
3	文字列操作	6
4	小数	7

第I部

はじめに

1 競プロとは

競技プログラミング(以下、競プロ)とは、名前の通りプログラミングを用いた競技である。詳細を、Wikipedia[1] から引用する。

競技プログラミングでは、参加者全員に同一の課題が出題され、より早く与えられた要求を満足するプログラムを正確に記述することを競う。コンピュータサイエンスや数学の知識を必要とする問題が多い。多くのコンテストでオンラインジャッジが採用されている。

また、競技プログラミングに参加する人を「競プロ er」と呼ぶことがある。

1.1 魅力

我々競プロ er が競プロに取り組む理由は、次の通りである [3]。

1. プログラミングスキル向上
2. アルゴリズムの習得
3. 数学的考察力の向上
4. 大人数とリアルタイムで競える

これらが、代表的な魅力だ。

幣学の学生の多くは、プログラミングに興味ある者が多いように思われる。電気通信大学では、1年次の後期の必修科目で「Ruby」と「C言語」を学ぶ。また、今年度から前期の「総合コミュニケーション科学」にて、「Python」にも触れる。うれしいことに、プログラミングに触れる機会は必ず訪れる。しかし、講義だけでモチベーションは維持できるだろうか。演習量は、足りるだろうか。プログラミングの学習がより楽しいものになれば、モチベーションがもっと上がるはずだ。競プロの最大の魅力は、自分の書いたコードで、ライバルと競い合う「ゲーム」性にあると思う。このゲーム性が一番面白いといっても過言ではない。

競プロで競うポイントを挙げる。

1. 正確か正確でないか
2. 解答速度
3. 実行時間
4. メモリの使用量

以上の点を、同じ言語同士、あるいは異なる言語で同士で競い合う。

正確か正確でないかは、正しいかどうかということだ。試験での丸かバツかということでもある。解答速度とは、解き始めてから解き終わる (解答コードを提出) までの所要時間だ。ここが競プロで重要な評価基準である。実行時間とメモリの使用量については、コンテストでは一部の例外を除いて、評価対象ではない。

1.2 AtCoder における実力評価

近年のコンテスト参加者は国内外で、増加傾向にある。世界最高峰の競技プログラミングサイト「AtCoder」では、コンテストにおける実力を次のように評価している。情報元は AtCoder 社の代表である chokudai 氏のブログだ [2]。

灰 参加すれば誰でもなれるので意欲以外の保証はなし。

茶 学生で茶色なら優秀だがエンジニアとしてはちょっと物足りない。派遣とかで来たエンジニアが茶色あれば一安心。

緑 大抵の企業でアルゴリズム力は十分。AtCoder 的には決して上位ではないが、他社評価サイトなら最高評価。

水 基礎的なアルゴリズム処理能力については疑いのないレベル。

青 青以上は一部上場の IT 企業でも、一人もいないことが結構あるレベルになる。

黄 黄色からは化け物。競プロの問題を解く機械だと思っておけば良い。

橙 あたまおかしい。

赤 もうなんか世界大会とかに招待されたりする。

Color	Rank	Rating	上位%
赤	SSS	2800 - 3199	0.3
橙	SS	2400 - 2799	1
黄	S	2000 - 2399	3
青	A	1600 - 1999	7
水	B	1200 - 1599	15
緑	C	800 - 1199	30
茶	D	400 - 799	50
灰	E	1 - 399	100

2 AtCoder の始め方

1. アカウント作成
2. 言語を決める。こだわりがないなら、「C++」を勧める。
3. C++を使うなら、『C++入門 AtCoder Programming Guide for beginners (APG4b)』(<https://atcoder.jp/contests/APG4b>)でプログラミングを学習
4. 『AtCoder Beginners Selection』(<https://atcoder.jp/contests/abs>)を解く
5. 『AtCoder (競技プログラミング) の色・ランクと実力評価、問題例』(<http://chokudai.hatenablog.com/entry/2019/02/11/155904>)をさらっと読む。完璧な理解は不要。
6. 過去の「AtCoder Beginner Contest」の A, B, C 問題を解く。
7. 「AtCoder Beginner Contest」に参加

「C++」を勧めるのは、AtCoder での利用者が大多数であることと、その実行速度の速さである。ネットで調べる際も、「C++」の記事が目立つ。既に経験のある言語がないなら、お勧めする。

『C++入門 AtCoder Programming Guide for beginners (APG4b)』でプログラミングを学習とあるが、一度で完璧にマスターする必要はない。困った時や時間が経ってから、再度復習するつもりで、さらっと済ませても良いと思われる。

『AtCoder Beginners Selection』は、直ぐにとは言わないが、全部解けるようになって欲しい問題だ。APG4b の 2 章が済んだら、解ける問題からやり始めると良いだろう。

『AtCoder (競技プログラミング) の色・ランクと実力評価、問題例』を読むと、コンテスト及び AtCoder の、実力評価の指標である Rating について分かる。完璧な理解は必要なく、簡単に読んで頂けたら充分だ。

何よりも Rating を上げるには、コンテストに参加するのが一番だ。「AtCoder Beginner Contest」に積極的に参加しよう。初めは、A, B 問題を完答することを目標にしよう。加えて過去のコンテストの問題を時間があるときに解いておくといよい。上級者ほど、過去問を解いている人は多く、新しく入った方でも、過去問を解くことでみるみる実力が上がっていくので、ぜひ取り組んでほしい。

3 進め方

1 回目は、C++ と Ruby の紹介をする。2 つの言語を体感し、自分に合いそうな言語を探してもらいたい。既に自分の言語が決まっている方は、その言語でも差し支えない。講習内容が既知であったら、次の問題を解いておくことを勧める。

2 回目では、1 回目で学んだこと事項で、実際に過去問を解いていきます。

第II部

Python

maccha

1991年に誕生したプログラミング言語。静的型付けのC言語とは違った、動的型付けという特性をもつ。機械学習など、近年注目を浴びている言語。Python3の台頭によりPython2からの移行が進みつつある。ここでWikipedia[2]からの引用を掲載する。

Pythonは、汎用のプログラミング言語である。コードがシンプルで扱いやすく設計されており、C言語などに比べて、さまざまなプログラムを分かりやすく、少ないコード行数で書けるといった特徴がある。

1 はじめに

Pythonは改行時に「;(セミコロン)」は必要ない。また、変数を使う前に型を宣言する必要がなく¹、コーディング上でも型を強く意識しない状況もある。四則演算の記述方法は、C言語と共通するが、除算「x / y」については注意がある。C言語ではint型同士の除算は、演算結果もint型となる²が、Pythonは型に関わらず、除算の演算結果はfloat型になる。この結果は、商ではない。割り算の商を得る演算は、「x // y」と記述する。

2 if

下記のコードは、入力で得た整数nが正なら”Positive”、0なら”Zero”、それ以外(負)なら”Negative”と出力するPythonのプログラム。

Listing 1: Python.if

```
n = int(input())
if n > 0:
    print("Positive")
elif n == 0:
    print("Zero")
else:
    print("Negative")
```

一方、C言語だとこうなる。

Listing 2: C.if

```
int main(){
    int n;
    scanf("%d", &n);
    if(n > 0){
```

¹変数や関数の返り値などの値の型が、プログラム実行時に決まる性質、動的型付け

²割り算という商

```

    printf("Positive\n");
} else if(n == 0){
    printf("Zero\n");
} else {
    printf("Negative\n");
}
}

```

if 文の説明をする前に、Python の int, input 関数の説明をしたい。int 関数³は、引数である数値以下の、最大の整数を返す組み込み関数⁴である。例えば、小数 33.4 を引数として渡せば、33 が返される。

input 関数は、入力された文字列を返す組み込み関数である。返り値が string 型であることに留意すること。

では、if 文について説明する。C と Python の違いを述べる。

- 条件式を”()”で囲まない
- 条件式の末尾に”.”におく
- C 言語でいうブロックの中身 ({} の中身) は、直属の if 文から 1 回インデントする必要がある
- ”&&”ではなく、”and”
- ”||”ではなく、”or”
- ”else if”ではなく、”elif”

3 文字列操作

Python における文字列の方は str 型のみである。

Listing 3: Python_str

```

s = input()

s1 = "At"
s2 = "Coder"

s12 = s1 + s2

```

input で代入される情報は、str 型である。変数に、直接文字列を代入するには、3,4 行目のようにすればよい。文字列「”」で囲むのだ。さらに 6 行目のように、文字列を結合することも可能だ変数 s12 には、”AtCoder”という文字列が格納される。

³浮動小数点に変換するなら”float”、文字列なら”str”

⁴言語の仕様として、標準で使用できる関数。これに対して、ユーザ自身がコード上で定義する関数を「ユーザ定義関数」という。

4 小数

Python では、小数は float のみである。また、int 型と float 型との演算ができる。演算結果は float 型となる。

Listing 4: Python_float

```
i = 1
f = 1.0
g = float(i)
j = int(f)
```

1 行目 int 型

2 行目 float 型

3 行目 int 型から float 型に変換した

4 行目 float 型から int 型に変換した

参考文献

- [1] ©AtCoder, 『AtCoder：競技プログラミングコンテストを開催する国内最大のサイト』 (<https://atcoder.jp>), 最終閲覧日 2020/6/7
- [2] Wikipedia, 『Python』 (<https://ja.wikipedia.org/wiki/Python>), 最終閲覧日 2020/6/7
- [3] Python Software Foundation, 『Python 3.8.3 ドキュメント』 (<https://docs.python.org/ja/3/>), 最終閲覧日 2020/6/7
- [4] Al Sweigart 著, 相川愛三 訳, 『退屈なことはPythonにやらせよう - ノンプログラマーにもできる自動化処理プログラミング』, オライリー・ジャパン, 2019

参考文献

- [1] Wikipedia, 『競技プログラミング』 (<https://ja.wikipedia.org/wiki/競技プログラミング>), 最終閲覧日 2020/6/20
- [2] chokudai(id:chokudai), 『AtCoder（競技プログラミング）の色・ランクと実力評価、問題例』 (<http://chokudai.hatenablog.com/entry/2019/02/11/155904>), 最終閲覧日 2020/6/20
- [3] E869120, 『レッドコーダーが教える、競プロ・AtCoder 上達のガイドライン【初級編：競プロを始めよう』 (<https://qiita.com/e869120/items/f1c6f98364d1443148b3>), 最終閲覧日 2020/6/20