

2023年11月1日



## AOJ 3.0 チュートリアル

<https://onlinejudge.u-aizu.ac.jp/beta/ice/>

UNIVERSITY OF AIZU

## 目次

はじめに .....	2
アカウント登録・ログイン .....	3
基本操作 .....	4
設定 .....	5
問題を選択する .....	6
コードを書く .....	9
テストする .....	10
AOJ へ提出する .....	12
解説を閲覧する .....	14
過去の提出履歴を閲覧する .....	16
模範解答を閲覧する .....	17
ブックマーク .....	19
前の問題・次の問題 .....	20
問題を検索 .....	21
ダッシュボード .....	22
他ユーザーのダッシュボードを表示 .....	24
ショートカットキー .....	25
お問い合わせ .....	25

## はじめに

---

このチュートリアルでは、AOJ 3.0 の使い方を解説します。今後のリリースへ向けてユーザーインターフェースや機能が変更される可能性がありますので、あらかじめご了承ください。

AOJ 3.0 はオンラインコーディングエディタとして実装されています。AOJ 3.0 を通して、以下の作業を画面ひとつで行うことができます。

- 問題の検索
- 問題文の閲覧
- コーディング
- サンプル入出力や指定された入出力を用いた動作確認
- プログラムの提出と判定結果確認
- 学習の進捗を確認
- その他

## アカウント登録・ログイン

以下の URL から AOJ 3.0 にアクセスしてみましょう。AOJ 3.0 の全ての機能を使用するには AOJ のアカウントでシステムにログインする必要があります。

<https://onlinejudge.u-aizu.ac.jp/beta/ice/>



サイトにアクセスした後、

1. 左上のメニューから「ユーザー」→「ログイン」を選択してください。
2. ポップアップが表示されるので、
  - ① アカウントをお持ちでない方は「アカウント新規作成」から AOJ のアカウントを作成してください。
  - ② アカウントをお持ちの方は、ご自分のユーザー ID とパスワードを入力し、ログインしてください。



## 基本操作

画面中央にあるバーをクリックしながら左右に移動させることで、パネルの幅を変更できます。またテキストエディタ下部のバーをクリックしながら上下に移動させると、エディタの高さを変更できます。



エディタ下部の拡大・縮小ボタンをクリックすると、エディタのフォントサイズを変更できます。



## 設定

「ユーザー」→「設定」からサイトの設定を変更できます。



設定画面ではショートカットキーの変更とダークモードへの切り替えが可能です。

「ジャッジステータスの通知を表示」では、「試す」・「提出」の際に、通知を表示するかどうか変更できます。



## 問題を選択する

問題を検索して、問題文を開いてみましょう。

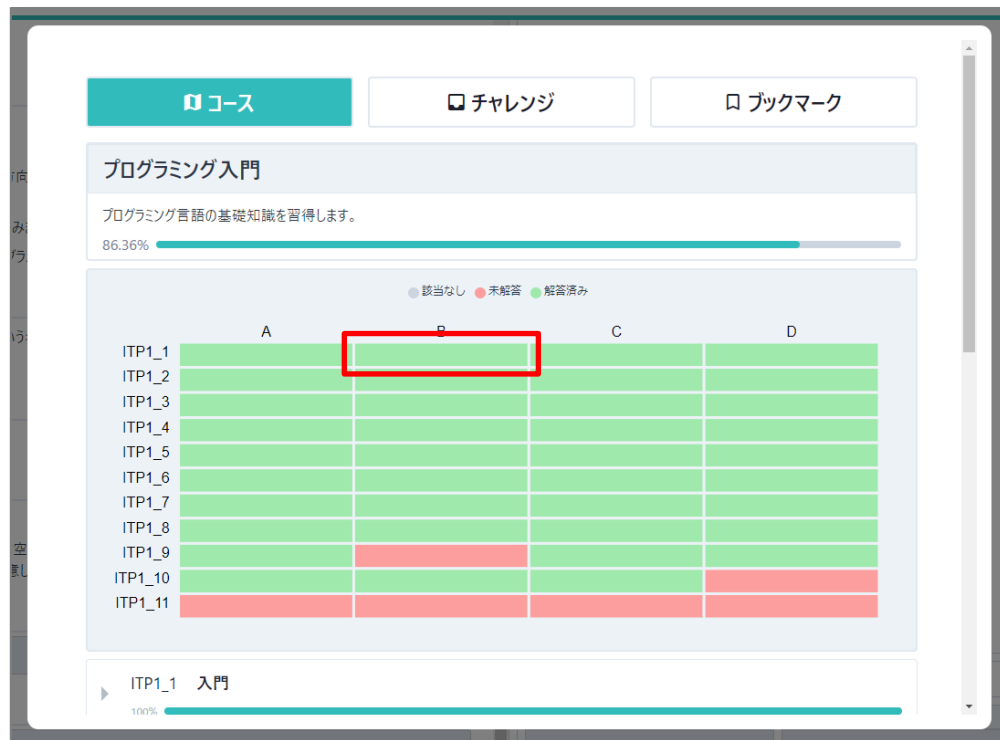
1. 「問題集」→「コース」・「チャレンジ」・「ブックマーク」・「アリーナ」いずれかをクリックすると、問題検索パネルが開きます。



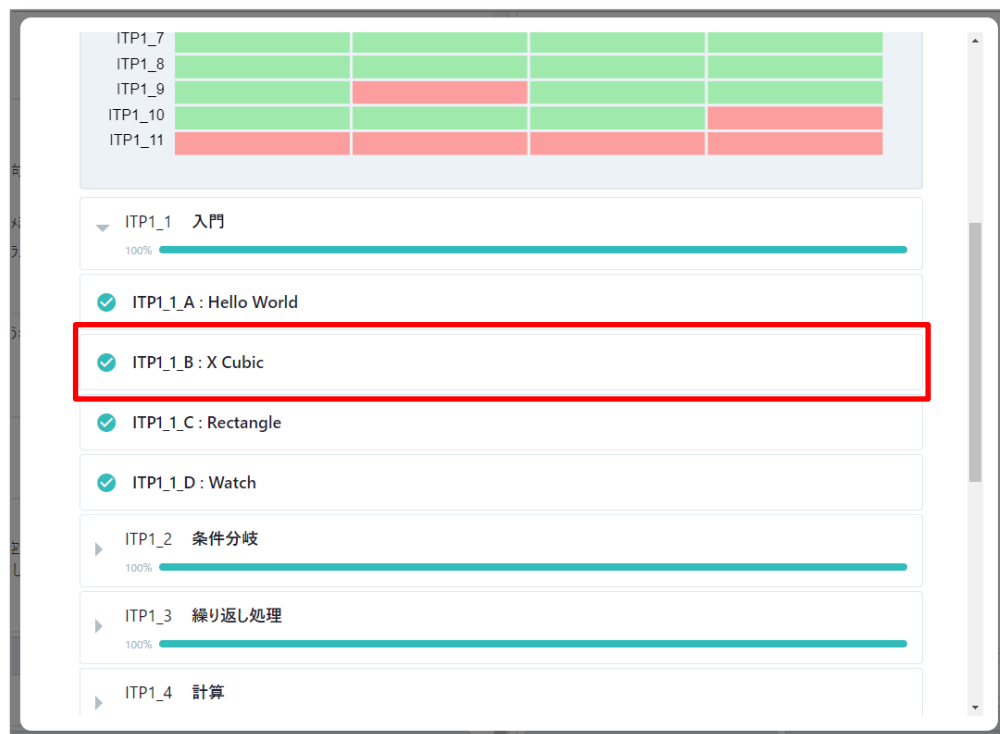
2. 「コース」を選んでみましょう。各コースはカードで表されています。各カードに表示されているプログレスバーは、そのコースの進捗状況を示します。



3. 続いて「プログラミング入門」を開いてみましょう。コースは複数のトピックから構成されています。ヒートマップや各トピックのプログレスバーから進捗を確認できます。



4. ヒートマップ上のセル「ITP1\_1\_B」または、「ITP1\_1 入門」→「X Cubic」を開いてみましょう。



5. 左側のパネルに問題文が表示されます。右側のパネルにはテキストエディタが表示され、下部のコンソールには選択した問題のサンプルケースが自動的に入力されます。

The screenshot displays the AOJ 3.0 interface for the problem "xの3乗". The left sidebar contains navigation options like "統計", "解説(前)", "問題文", "解説(後)", "提出履歴", and "模範解答". The main content area shows the problem title "xの3乗", a 1-second and 131072 KB limit, and instructions to write a program that calculates the cube of an integer x. The "Input" section states that one integer x will be provided. The "Output" section requires the result to be printed on one line. The "Constraints" section specifies  $1 \leq x \leq 100$ . Below, "Sample Input 1" shows the value 2, and "Sample Output 1" shows the value 8. The right panel includes a code editor with a single line of code, a "ジャッジの詳細" button, and a "最近の提出" button. At the bottom, there are buttons for "試す" and "提出", and a console area showing the input 2 and the expected output 8.

## コードを書く

テキストエディタでコードを書いてみましょう。

The screenshot shows the AOJ 3.0 problem page for "xの3乗". The page layout includes a top navigation bar with the problem ID "ITP1\_1\_B", a left sidebar with navigation options like "統計" (Statistics) and "問題文" (Problem Statement), and a main content area. The main content area displays the problem title "xの3乗", a description, input/output requirements, and constraints. A red box highlights the code editor area, which contains the number "1". Below the editor, the "アップロード" (Upload) button is also highlighted with a red box. The interface includes a navigation menu on the left, a top navigation bar with the problem ID "ITP1\_1\_B", and a right sidebar with utility buttons.

赤い枠で示した部分にプログラムのコードを書きます。または、エディタ下部のアップロードボタンでエクスプローラーを開き、ローカルファイルシステムからソースコードを読み込むこともできます。加えて、ダウンロードボタンから、エディタ上のソースコードをローカルファイルシステムへ保存することができます。

## テストする

コーディングが終わったら、テスト機能でコードが正しく動作するか確認しましょう。テストはサンプル入出力を用いた動作確認を行うための、簡易的なチェック機能です。



The screenshot shows the AOJ 3.0 online judge interface. At the top, there is a header with the problem ID 'P1\_1\_B', a search bar, and language selection options for 'JAVA' and 'Japanese'. The main area is a code editor containing the following Java code:

```
1 import java.io.BufferedReader;
2 import java.io.IOException;
3 import java.io.InputStreamReader;
4
5 class Main {
6
7     public static void main(String args[] throws IOException {
8         BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
9         int x = Integer.valueOf(in.readLine());
10        x = x * x * x;
11        System.out.println(x);
12    }
13 }
14
15
16
17
18
19
20
21
22
23
24
25
```

Below the code editor, there is a control panel with buttons for '試す' (Test) and '提出' (Submit). A red arrow points to the '試す' button. Below the control panel, there is a table with three columns: '入力' (Input), '想定出力' (Expected Output), and '出力' (Output). The '入力' column contains the value '2', and the '想定出力' column contains the value '8'. The '出力' column is empty. A red box highlights the '入力' and '想定出力' columns.

1. 「入力」エリアと「想定出力」エリアにそれぞれ、プログラムが読み込む入力値、期待するプログラムの出力値を設定します（あらかじめサンプル入出力が設定されています）。
2. 「試す」をクリックすると、ここで設定された入力値・出力値とともにプログラムが送信されます。結果が返ってくるまでは、時間がかかる場合があります。
3. コンパイルと実行が成功すれば、「出力」エリアに、プログラムの出力が表示されます。コンパイルや実行に失敗すると、エラーメッセージが表示されます。

ジャッジは、以下のいずれかのステータスを返します。

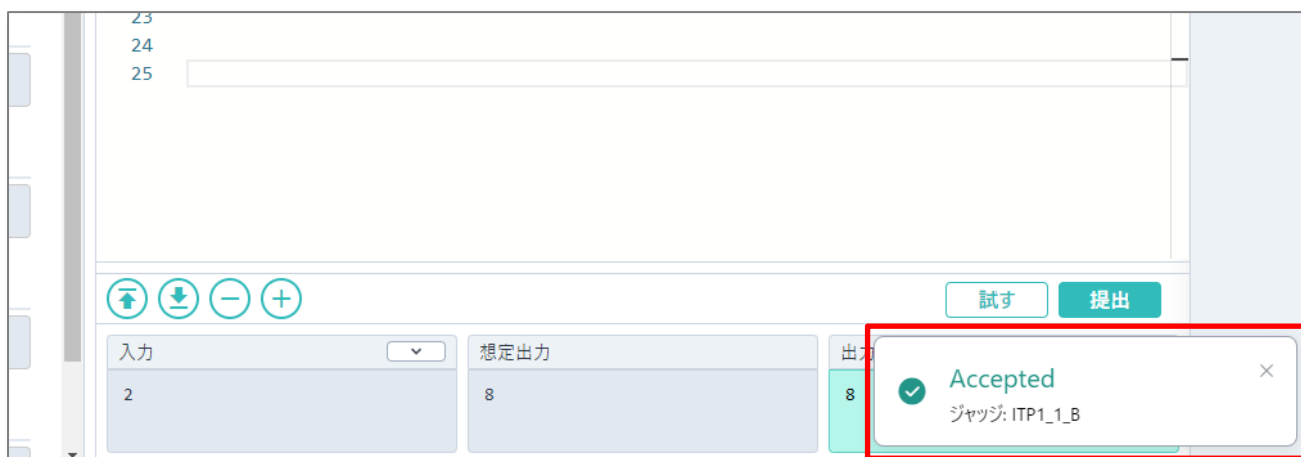
<b>AC</b> <b>正解</b> (Accepted)
<b>WA</b> <b>不正解</b> (Wrong Answer) <ul style="list-style-type: none"><li>・ ジャッジデータが想定した出力とは異なる出力が行われた</li></ul>
<b>PE</b> <b>表示形式に誤り</b> (Presentation Error) <ul style="list-style-type: none"><li>・ 余計な空白や改行がある、あるいは必要な空白や改行がない</li></ul>
<b>CE</b> <b>コンパイルエラー</b> (Compile Error) <ul style="list-style-type: none"><li>・ 提出されたプログラムのコンパイルに失敗した</li></ul>
<b>TLE</b> <b>時間制限超過</b> (Time Limit Exceeded) <ul style="list-style-type: none"><li>・ 問題で指定された制限時間内にプログラムが終了しなかった</li></ul>
<b>MLE</b> <b>メモリ制限超過</b> (Memory Limit Exceeded) <ul style="list-style-type: none"><li>・ 問題で指定された以上のメモリを使用した</li></ul>
<b>RE</b> <b>実行時エラー</b> (Runtime Error) <ul style="list-style-type: none"><li>・ 提出されたプログラムの実行中にエラーが発生した</li></ul>

## AOJ へ提出する

テストで正しい結果が得られたら AOJ に提出しましょう。



「提出」ボタンからコードを提出します。提出されたプログラムは、ジャッジデータによって厳格に採点されます。提出結果はサイト下部に通知されます。



ステータスを詳しく確認したい場合は、メニューの「ジャッジの詳細」アイコンをクリックします。テストケースごとの詳しいジャッジの結果を確認することができます。

画面右端の「貼り付け」ボタンをクリックすると、エディタ下部の「入力」エリアと「想定出力」エリアにそれぞれ選択したテストケースの「入力」・「出力」が貼り付けられます。

## 解説を閲覧する

解答プログラムの書き方が分からない場合や、模範解答を知りたい場合は解説を参考にしましょう。メニューの「解説（前）」と「解説（後）」から、解説を表示できます。

「解説（前）」では、主に問題文を読む前に役立つ、文法や方針の説明を閲覧することができます。

The screenshot shows the AOJ 3.0 Tutorial interface. At the top, there are navigation menus for 'ユーザー', '問題集', 'リンク集', and 'ヘルプ', along with a 'ITP1' status indicator. The left sidebar contains several menu items: '統計', '解説(前)', '問題文', '解説(後)', '提出履歴', '模範解答', and '前の問題'. The '解説(前)' and '解説(後)' items are highlighted with red boxes. The main content area displays the 'Pre note: 言語解説' for 'JAVA'. The content includes sections for '変数' (Variables) and '代入演算' (Assignment Operations).

**変数**

コンピュータ（プログラム）に入力された値は一時的にメモリに記録する必要があります。メモリの特定領域を確保して、指定した名前前でアクセスするための仕組みが**変数**です。

変数は、そのデータ型（整数、浮動小数点、文字、など）と名前の組で**宣言**をする必要があります。たとえば、1つの整数データを保持することができるxという名前の変数は以下のように宣言します。

```
int x;
```

**代入演算**

変数の役割は、プログラムで扱う値をメモリへ書き込み、必要な時に参照することです。最も基本となる**代入演算**では=の左側の変数（メモリ領域）に、右側の式の計算結果が書き込まれます。変数はその名の通り「変わる数」で、代入演算によって値を何度でも変更することができます。一方、式の中に含まれる変数は、それが指すメモリ領域の値を返します。たとえば、次のプログラムは、変数a, bにそれぞれ5, 8を格納し、その後それらの中身を交換するプログラムです。

```
int a, b, t; // 3つの整数を宣言
a = 5;      // a に 5 を格納
b = 8;      // b に 8 を格納
```

「解説（後）」では、解答後に役立つ、模範解答や詳しい説明を閲覧することができます。



The screenshot shows the AOJ 3.0 Tutorial interface. The top navigation bar includes 'ユーザー', '問題集', 'リンク集', 'ヘルプ', and 'ITP1'. The left sidebar contains navigation icons for '統計', '解説(前)', '問題文', '解説(後)', '提出履歴', '模範解答', and '前の問題'. The main content area displays two sections:

- Post note: 一般解説** (Challenge): Includes a link for '演習問題' with a sub-link for 'ワード'.
- Post note: 言語解説** (JAVA): Titled 'JAVAの模範解答', it contains a code block for a Java program:

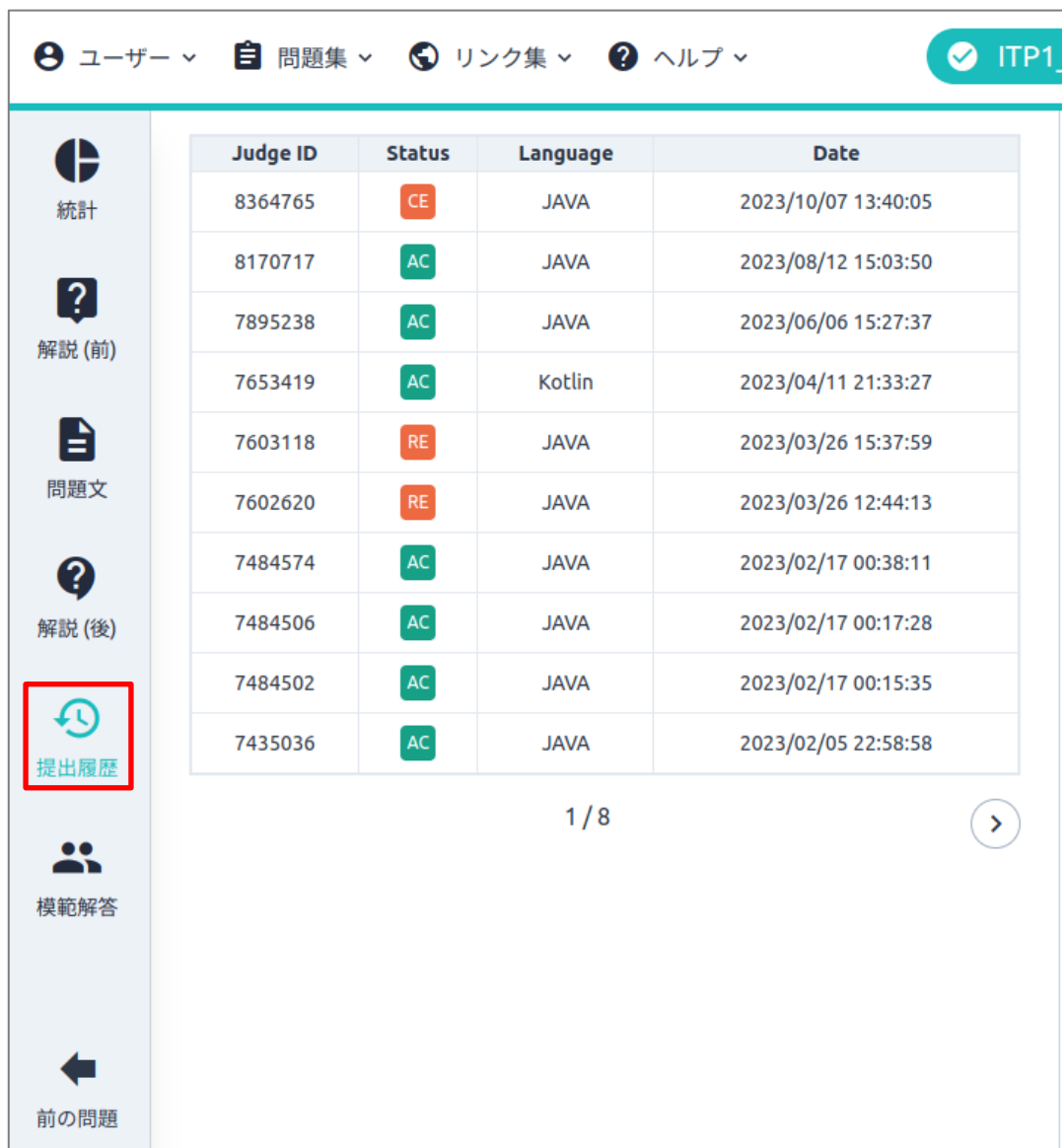
```
import java.util.Scanner;

class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int x = sc.nextInt();
        System.out.println(x * x * x);
    }
}
```

解説には、「言語解説」や「一般解説」などの種類があります。言語解説では、指定したプログラミング言語の文法に関する解説を閲覧することができます。一般解説では、想定するアルゴリズムやヒントを閲覧することができます。（※解説がサポートされていない問題もあります。予めご了承ください）

## 過去の提出履歴を閲覧する

過去に提出したソースコードを確認できます。メニューの「提出履歴」アイコンをクリックすると、選択中の問題に対する過去の提出履歴を表示することができます。

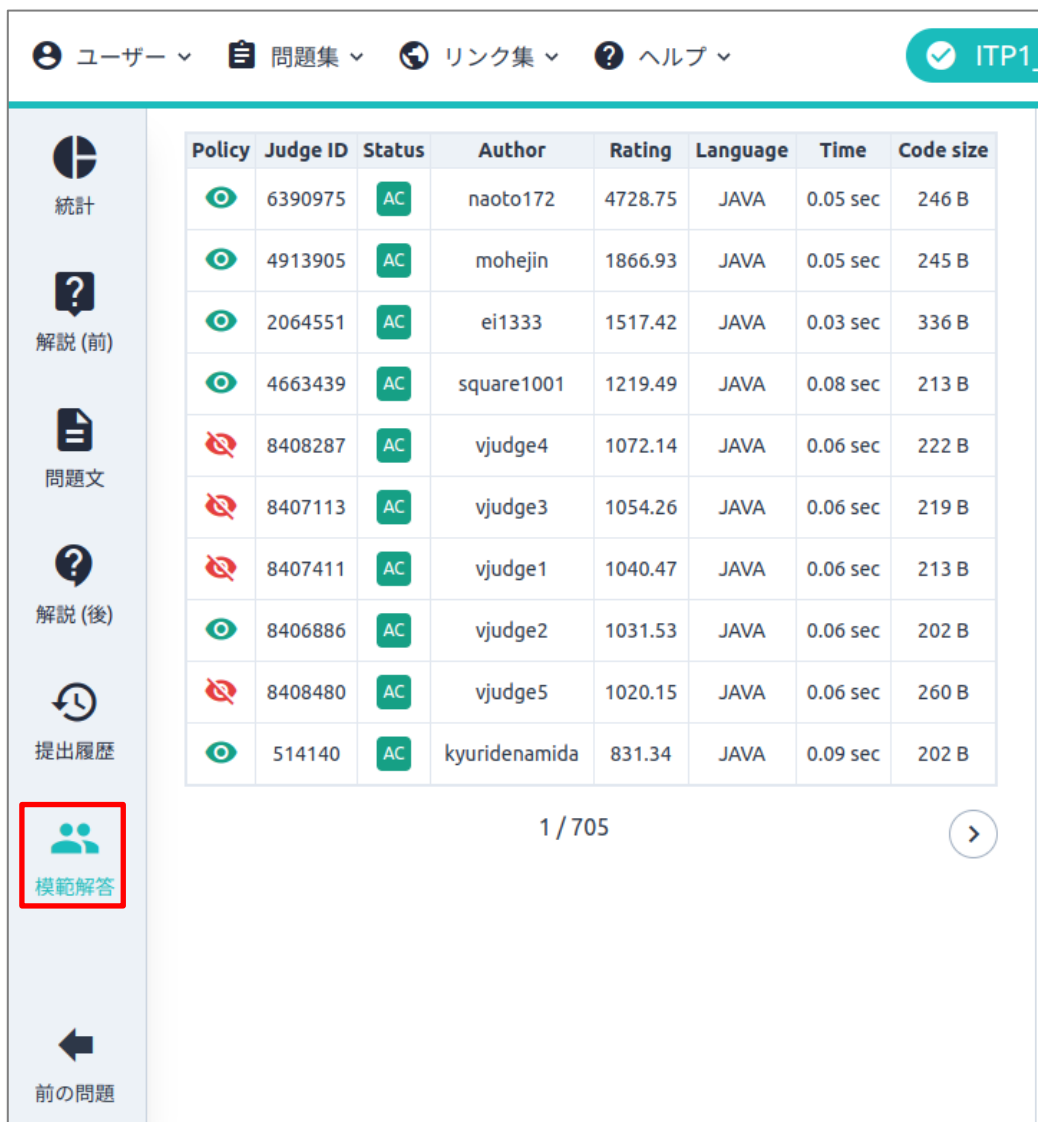


Judge ID	Status	Language	Date
8364765	CE	JAVA	2023/10/07 13:40:05
8170717	AC	JAVA	2023/08/12 15:03:50
7895238	AC	JAVA	2023/06/06 15:27:37
7653419	AC	Kotlin	2023/04/11 21:33:27
7603118	RE	JAVA	2023/03/26 15:37:59
7602620	RE	JAVA	2023/03/26 12:44:13
7484574	AC	JAVA	2023/02/17 00:38:11
7484506	AC	JAVA	2023/02/17 00:17:28
7484502	AC	JAVA	2023/02/17 00:15:35
7435036	AC	JAVA	2023/02/05 22:58:58

リストから特定の提出履歴をクリックすると、そのコードを閲覧することができます。

## 模範解答を閲覧する

他のユーザーが解答したコードを閲覧することができます。メニューの「模範解答」アイコンをクリックすると、選択中の問題に対する模範解答を表示することができます。



ユーザー ▾ 問題集 ▾ リンク集 ▾ ヘルプ ▾ ITP1\_

Policy	Judge ID	Status	Author	Rating	Language	Time	Code size
	6390975	AC	naoto172	4728.75	JAVA	0.05 sec	246 B
	4913905	AC	mohejin	1866.93	JAVA	0.05 sec	245 B
	2064551	AC	ei1333	1517.42	JAVA	0.03 sec	336 B
	4663439	AC	square1001	1219.49	JAVA	0.08 sec	213 B
	8408287	AC	vjudge4	1072.14	JAVA	0.06 sec	222 B
	8407113	AC	vjudge3	1054.26	JAVA	0.06 sec	219 B
	8407411	AC	vjudge1	1040.47	JAVA	0.06 sec	213 B
	8406886	AC	vjudge2	1031.53	JAVA	0.06 sec	202 B
	8408480	AC	vjudge5	1020.15	JAVA	0.06 sec	260 B
	514140	AC	kyuridenamida	831.34	JAVA	0.09 sec	202 B

1 / 705

統計  
解説 (前)  
問題文  
解説 (後)  
提出履歴  
**模範解答**  
前の問題

解答は各ユーザーのレーティングをもとに降順に表示されます。提出履歴と同様に、リストから模範解答をクリックすると、そのコードを閲覧することができます。「Diff」ボタンをクリックするとご自身のコードと模範解答のコードを比較することができます。

Judge ID: 5924926 By: yutaka

```

1 import java.util.Scanner;
2
3 class Main{
4     public static void main(String[] args){
5         Scanner sc = new Scanner(System.in);
6
7         int x = sc.nextInt();
8
9         System.out.println(x*x*x);
10    }
11 }
12

```

```

1+import java.io.BufferedReader;
2+import java.io.IOException;
3+import java.io.InputStreamReader;
4
5+class Main {
6
7+    public static void main(String args[])
8+        BufferedReader in = new BufferedRea
9+        int x = Integer.valueOf(in.readLine
10+        x = x * x * x;
11+        System.out.println(x);
12    }
13 }
14
15+
16+
17+
18+
19+
20+
21+
22+
23+
24+

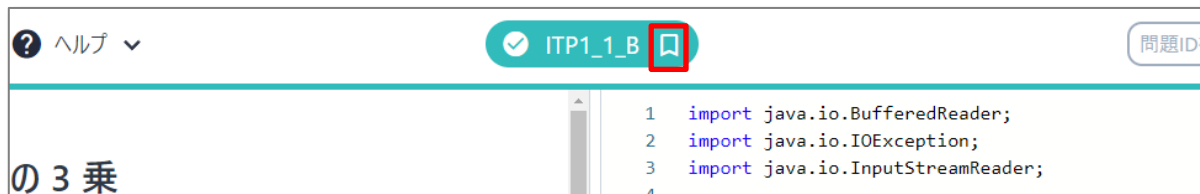
```

Diff コピー 閉じる

Policy	Judge ID	Status	Author	Rating	Language	Time	Code size
	2943892	AC	emtsu_ba	678.05	JAVA	0.06 sec	243 B

## ブックマーク

問題文にある「ブックマーク」ボタンをクリックすることで、現在の問題を保存できます。



ブックマークした問題は検索パネルから確認できます。



## 前の問題・次の問題

問題文にある「前の問題」「次の問題」ボタンをクリックすると、各コースの前の問題・次の問題が表示されます。

The screenshot shows the AOJ 3.0 problem page for "xの3乗". The page layout includes a top navigation bar with user, problem set, link set, and help menus. The main content area is divided into a left sidebar with navigation icons (統計, 解説(前), 問題文, 解説(後), 提出履歴, 模範解答) and a main content area. The main content area contains the problem title "xの3乗", a description, input/output sections, constraints, and sample input/output. The "前の問題" button is located in the left sidebar, and the "次の問題" button is located in the right sidebar. Both buttons are highlighted with red boxes.

問題IDを入力 決定 JAVA Japanese

1 sec 131072 KB

### xの3乗

1つの整数  $x$  を読み込んで、 $x$  の3乗を計算し結果を出力するプログラムを作成して下さい。

**Input**  
入力として1つの整数  $x$  が与えられます。

**Output**  
 $x$  の3乗の計算結果を1行に出力して下さい。最後の改行を忘れないようにしましょう。

**Constraints**  
Constraints (制約)には、採点に使用される入力データの上限などの情報が書かれています。変数の型を選ぶときや、アルゴリズムを考えるときに必要な情報になります。この問題に関する制約は以下の通りとなります：

- $1 \leq x \leq 100$

Sample Input 1  
2

Sample Output 1  
8

← 前の問題

次の問題 →

## 問題を検索

問題検索フォームに問題 ID を入力すると、その問題を開くことができます。問題 ID とはヘッダーに表示されている、各問題に割り当てられている ID を指します。



The screenshot shows the AOJ problem search interface. At the top, there are navigation links for 'リンク集' and 'ヘルプ'. The search bar at the top right contains the ID 'ITP1\_1\_D' and a '決定' (Submit) button. The breadcrumb trail shows 'ITP1\_1\_B'. The main content area displays the problem title 'x の 3 乗' and the description 'x の 3 乗を計算し結果を出力するプログラムを作成して下さい。'

## ダッシュボード

「ユーザー」→「ダッシュボード」をクリックすると、ユーザーの各ステータスがまとめられたダッシュボードを閲覧できます。



ダッシュボードには「ユーザー名」「提出数」「レーティング」などの基本的な情報が表示されます。加えて「直近1年のACヒートマップ」「毎週のレーティング」「言語別解答数」「問題別レーティング」等、より複雑な情報を閲覧できます。





検索フォームに他ユーザーの ID を入力すると、そのユーザーと比較したステータスが確認できます。



## 他ユーザーのダッシュボードを表示

---

他のユーザーのダッシュボードを表示するためには、WebブラウザのURLから、

「<https://onlinejudge.u-aizu.ac.jp/beta/ice?userId=ユーザーID>」 のように指定することでダッシュボードを表示できます。例えばユーザーIDが aizutaro の場合、

「<https://onlinejudge.u-aizu.ac.jp/beta/ice?userId=aizutaro>」 と指定します。

指定した URL にアクセスすると、他のユーザーのダッシュボードが表示されます。

## ショートカットキー

---

AOJ 3.0 では以下のショートカットキーをサポートしています。

- Alt+Shift+X: 試す (デフォルト)
- Alt+X: 提出 (デフォルト)
- Ctrl+P: コースパネルを表示
- Ctrl+` : チャレンジパネルを表示
- Ctrl+{ : ブックマークパネルを表示
- Ctrl+M: ダッシュボード表示
- Ctrl+F: 次の問題
- Ctrl+D: 前の問題
- Ctrl+1, 2 ...9, 0: サンプルケースを選択
- Ctrl+I: 右側パネルの選択を上へ移動
- Ctrl+K: 右側パネルの選択を下へ移動
- Ctrl+U: 左側パネルの選択を上へ移動
- Ctrl+J: 左側パネルの選択を下へ移動
- Ctrl+Shift++: フォント拡大
- Ctrl+-: フォント縮小

## お問い合わせ

---

本システムは現在開発中です。バグのご報告や、ご提案・ご要望は [uoaojs@gmail.com](mailto:uoaojs@gmail.com) または [yutaka@u-aizu.ac.jp](mailto:yutaka@u-aizu.ac.jp) までご連絡ください。